Accounting en
Accounting protocollen

Arthur de Jong

onderzoeksverslag

2000−02−11

West Consulting BV
Bagijnhof 80
2611 AR Delft

Technische Universiteit Delft
Faculteit Informatietechnologie en Systemen
Afdeling Information Systems & Software Engineering i.o.
Groep Software Engineering, Programmeren, Programmeertalen & Compilers

Afstudeergegevens

Titel:	Accounting en Accounting protocollen
Schrijver:	Arthur de Jong <arthur@ch.twi.tudelft.nl>
Begeleiders:	Frans Ververs <f.ververs@its.tudelft.nl> (TUDelft)
	Leo Root <leonard@west.nl> (West Consulting BV)

Korte samenvatting:

Accounting is het verzamelen van gegevens over gebruik van diensten. Bij het transporteren van deze gegevens komen accounting protocollen om de hoek kijken. Voor het transport van accounting gegevens bij bijvoorbeeld inbellen zijn verschillenden protocollen ontwikkeld. Er is echter behoefte aan een accounting protocol dat toepasbaar is in meerdere gebieden. Accounting wordt toegepast bij allerlei vormen van digitale dienstverlening, zoals inbellen, telefonie, application hosting en het bieden van quality of service verbindingen. Om een accounting protocol algemeen toepasbaar te maken moet het voldoen aan een groot aantal requirements. Het blijkt dat er nog geen protocol is dat aan alle requirements voldoet. Wel zijn aspecten van bestaande protocollen bruikbaar voor het ontwerp van een nieuw protocol. Dit verslag geeft een beschrijving van accounting, behandelt haar toepassingen, somt de requirements voor een algemeen accounting protocol op, licht kort enkele bestaande protocollen toe en vergelijkt deze. Aan de hand hiervan worden conclusies getrokken en aanbevelingen gedaan voor het gebruik van accounting protocollen en het ontwerp ervan.

Voorwoord

Binnen de Internet Engineering Task Force is men geruime tijd bezig met onderzoek naar accounting protocollen. Er is vraag naar een standaard voor het transporteren van accounting gegevens. Er zijn vanuit verschillende organisaties initiatieven om een protocol te ontwikkelen dat algemeen toepasbaar is.

Accounting op zich is geen nieuw vakgebied. Er komen echter veel nieuwe toepassingen met de groei van het internet en elektronische dienstverlening. Veel nieuwe diensten met uiteenlopende eigenschappen vragen om een protocol dat kan worden gebruikt om accounting bij deze diensten uit te voeren. Momenteel zijn verschillende onderzoeksgroepen bezig met AAA protocollen, wat maakt dat het een vakgebied is dat zich momenteel in een ontwikkelingsfase bevindt.

In het kader van een onderzoekstaak voor een afstudeerwerk heb ik een verslag geschreven over accounting en wat er zoal bij komt kijken. Over dit onderwerp wordt veel gepubliceerd en gediscussieerd en vooral op het internet is hierover veel te vinden. Al het materiaal voor dit onderzoek is dan ook van het internet afkomstig.

Speciaal wil ik graag Hella bedanken die mij heeft geholpen om met mijn belabberde Nederlands toch nog een leesbaar verhaal te maken.

2000−02−11
Arthur

Samenvatting

Bij het aanbieden van een dienst zijn authenticatie, autorisatie en accounting (AAA) van belang. Hierbij is accounting het verzamelen van gegevens over gebruik van diensten voor het kunnen maken van trenanalyses, het uitvoeren van auditing, het sturen van rekeningen (billing) of het plaatsen van kosten (cost allocation). Accounting wordt in het algemeen toegepast om inzicht te krijgen in het gebruik van diensten.

Toepassingen van accounting zitten in allerlei vormen van digitale dienstverlening, zoals inbellen, telefonie, application hosting en het bieden van quality of service verbindingen. In veel gevallen moet er voor het gebruik van de dienst betaald worden of zijn er kosten mee gemoeid. Dit maakt dat er financiele belangen zij die vooral eisen stellen aan de betrouwbaarheid van de accounting gegevens.

Bij het transporteren van deze gegevens komt een accounting protocol om de hoek kijken. Accounting gegevens worden in het algemeen over een netwerk getransporteerd, van het apparaat wat de dienst verzorgt naar de server die het gebruik bijhoudt. Meestal zullen deze niet zover van elkaar verwijderd liggen en zich binnen dezelfde organisatie bevinden. Het kan echter ook voorkomen dat deze door verschillende organisaties beheerd worden. Accounting gegevens worden dan tussen twee instanties uitgewisseld (inter−domain accounting). Hier komt vaak beveiliging bij kijken.

Voor enkele veel voorkomende diensten zijn accounting protocollen beschikbaar (vaak als onderdeel van een AAA protocol). Het blijkt echter dat deze protocollen vaak niet voldoen in inter−domain toepassingen. Ook voldoen deze protocollen vaak niet voor andere diensten. Omdat er veel nieuwe vormen van dienstverlening ontwikkeld worden is er behoefte aan een protocol wat algemeen toepasbaar is.

Om een accounting protocol algemeen toepasbaar te maken moet het voldoen uit een groot aantal requirements. Deze requirements zijn onder te verdelen in algemene requirements die voor het gehele accounting protocol gelden, eisen aan de beveiliging, de benodigde uit te wisselen berichten, requirements aan het transport van accounting gegevens en requirements die voor een record format gelden.

Het blijkt dat er nog geen accounting protocol is wat aan alle requirements voldoet. Het verdient aanbeveling om een eventueel nieuw protocol te ontwerpen als een verzameling van verschillende onderdelen. Hierbij zijn er aspecten van bestaande protocollen wel bruikbaar.

Inhoudsopgave

Afstudeergegevens ..... **iii**

Voorwoord ..... **v**

Samenvatting ..... **vii**

1. Inleiding **1**

2. Accounting **3**

2.1 Accounting protocol 3

2.2 Inter−domain accounting 5

2.3 Overhead 6

2.4 Push vs Pull 7

3. Toepassingen **9**

3.1 Inbellen 9

3.2 Telefonie 9

3.3 Telefooncentrale 9

3.4 Interne netwerkelementen 9

3.5 Website hosting 10

3.6 Application hosting 10

3.7 Roaming 10

3.8 Resource reservation 11

3.9 Overig 11

4. Requirements **13**

4.1 Algemene requirements 13

4.1.1 Real−time accounting (MUST) 13

4.1.2 Archival accounting (MUST) 13

4.1.3 Batch accounting (MUST) 14

4.1.4 Minimale overhead (SHOULD) 14

4.1.5 Schaalbaar (MUST) 14

4.1.6 Ondersteuning van eindige sessies (MUST) 15

4.1.7 Ondersteuning van oneindige sessies (MUST) 15

4.1.8 Ondersteuning van ondeelbare events (MUST) 15

4.1.9 Inter−domain accounting (MUST) 15

4.1.10 Meerdere accounting servers (MUST) 15

4.1.11 Samengestelde diensten (SHOULD) 15

4.2 Security requirements 16

4.2.1 Integrity protection (MUST) 16

4.2.2 Authenticatie (MUST) 16

4.2.3 Confidentiality protection (MUST) 17

4.2.4 Replay protection (MUST) 17

4.2.5 Non−repudiation (SHOULD) 17

4.2.6 Brokers (MUST) 17

4.3 Accounting event requirements 17

4.3.1 Start of a session (start bericht) (MUST) 17

4.3.2 End of a session (stop bericht) (MUST) 18

4.3.3 Update of a session (interim bericht) (MUST) 18

4.3.4 Session record (MUST) 18

4.3.5 Polling (MUST) 18

4.3.6 Event−driven polling (MUST) 18

4.3.7 Bevestiging van bericht (MUST) 19

4.3.8 Negotiation of transfer method and format capabilities (MUST) 19

4.4 Transport requirements 19

4.4.1 Betrouwbaar transport (MUST) 19

4.4.2 Ondersteuning grote berichten (MUST) 19

4.4.3 Snel van server veranderen (MUST) 19

4.4.4 Buffering van accounting gegevens (MUST)....20
4.4.5 Bidirectionele communicatie (MUST)....20
4.4.6 Flow control (MUST)....20
4.5 Record format requirements....20
4.5.1 Tagged and typed data (MUST)....20
4.5.2 Standaard datatypen (MUST)....20
4.5.3 Extensible (MUST)....21
4.5.4 Gegroepeerde of gestructureerde attributen (MAY)....21
4.5.5 Human readable (MAY)....21
4.5.6 Compact record format (SHOULD)....21
4.5.7 Uitbreidbare berichten (MUST)....21
4.5.8 Verschillende diensten (MUST)....22
4.5.9 Definitie diensten (SHOULD)....22
4.5.10 Samengestelde diensten (MUST)....22
5. Protocollen....23
5.1 RADIUS....23
5.2 DIAMETER....26
5.3 COPS....28
5.4 TACACS+....28
5.5 SNMP....30
5.6 MSIX....31
5.7 ADIF....33
5.8 TIPHON....34
5.9 OFX....35
6. Evaluatie protocollen....37
6.1 RADIUS....37
6.2 DIAMETER....37
6.3 TACACS+....37
6.4 SNMP....37
6.5 MSIX....38
6.6 ADIF....38
6.7 Overzicht....38
7. Conclusies en aanbevelingen....41
Literatuurlijst....43

1. Inleiding

Doel van dit verslag is inzicht geven in accounting en accounting protocollen. Het is gemaakt om als basis te dienen voor het kiezen van een accounting protocol voor het maken van een implementatie. Verder is het te gebruiken als inleiding in accounting en het gebruik van accounting protocollen.

Bij het aanbieden van diensten of middelen is het belangrijk inzicht te krijgen in het gebruik ervan. Accounting is het verzamelen van gegevens over gebruik met als doel het kunnen maken van een trendanalyse, het uitvoeren van auditing, het sturen van rekeningen (billing) of het plaatsen van kosten (cost allocation). Gegevens die hiervoor worden gebruikt zijn bijvoorbeeld gegevens over de duur van de sessie en identiteit van de client.

Bij het aanbieden van netwerkdiensten is het bijvoorbeeld wenselijk gegevens over de getransporteerde hoeveelheid data, gebruikte capaciteit en duur van een bepaalde sessie te meten. Deze accounting gegevens worden vaak centraal, in een accounting server opgeslagen. Om accounting gegevens naar de server te transporteren worden accounting protocollen gebruikt. Voor verschillende standaarddiensten zijn hiervoor protocollen ontwikkeld. Er is een behoefte aan standaardisatie van deze protocollen en een enkel uniform te gebruiken protocol, omdat accounting gegevens van verschillende diensten en door verschillende instanties worden uitgewisseld [blount−acct−service].

Als eerste zal een inleiding worden gegeven in accounting en accounting protocollen. De doelen van accounting zullen worden behandeld. Ook komen het gebruik van accounting protocollen en de opzet van accounting systemen aan de orde. Daarna zullen in hoofdstuk drie enkele toepassingen van accounting kort worden toegelicht. Hierdoor wordt een achtergrond voor accounting protocollen geschetst. In hoofdstuk vier zullen requirements aan accounting en accounting protocollen worden behandeld. Deze requirements kunnen worden gebruikt als basis voor het ontwerp van een accounting protocol en als evaluatiemiddel voor bestaande protocollen. In hoofdstuk vijf zullen een aantal veel gebruikte en nieuwe protocollen worden behandeld. Dit hoofdstuk is bedoeld om een globale indruk van het protocol te krijgen. In hoofdstuk zes worden de behandelde accounting protocollen vergeleken en aan de requirements getoetst. Als laatste zullen in hoofdstuk zeven enkele conclusies getrokken en aanbevelingen worden gedaan voor het gebruik van accounting protocollen en het ontwerp ervan.

2. Accounting

Service elements zijn elementen die een dienst aanbieden. Hierbij zijn authenticatie, autorisatie en accounting (AAA) van belang. Authenticatie is het vaststellen van de identiteit van een client (gebruiker van de dienst). Autorisatie is het bepalen of een client toegang heeft tot een dienst en met welke parameters. Aan autorisatie gaat in het algemeen authenticatie vooraf.

Het aanbieden van een enkele dienst door een service element aan een client wordt een sessie genoemd. Accounting is het verzamelen en transporteren van gegevens over gebruik van service elements. Deze gegevens worden in accounting attributen vastgelegd. Dit zijn onder andere gegevens over duur van de sessie, identiteit van de client, kwaliteit van het geleverde en andere zaken die van belang kunnen zijn. Voor accounting zijn authenticatie en autorisatie niet noodzakelijk, maar in een aantal gevallen wel wenselijk.

Het doel van accounting is het kunnen maken van trendanalyses, het uitvoeren van auditing, het sturen van rekeningen (billing) of het plaatsen van kosten (cost allocation). Hiermee wordt inzicht verschaft in het gebruik van diensten en wordt het aanbieden beheersbaar gemaakt.

Trendanalyse

Het doel van trendanalyse is het inzicht krijgen in gebruik en ontwikkelingen in gebruik van diensten. Met trendanalyse wordt een voorspelling van het gebruik van een bepaald service element in de toekomst gemaakt. Dit is nodig voor capaciteitsplanning of eventuele signalering van problemen of tekorten op de korte termijn. Trendanalyse legt niet al te strikte eisen aan accounting, omdat deze vaak niet real−time hoeft te worden uitgevoerd en omdat er vaak slechts een algemeen beeld van de gegevens nodig is om trendanalyse uit te voeren. Voor veel organisaties is het doen van trendanalyse wel van groot belang.

Auditing

Auditing heeft betrekking op het volgen van activiteiten van gebruikers. Met auditing kunnen bijvoorbeeld misbruik van middelen en diensten of zwakheden in de beveiliging worden opgespoord. Met auditing kan worden nagegaan of gebruikers zich aan de voorwaarden van hun gebruik houden. Er kunnen met behulp van policies grenzen worden gesteld aan gebruik van diensten, zowel over langere termijn als op korte termijn (piekbelasting). Verslaglegging alleen is niet voldoende om naleving van de policies af te dwingen. Er moet een (real−time) terugkoppeling plaatsvinden naar het autorisatie gedeelte om gebruik van de dienst te kunnen beperken.

Billing

Billing is het opstellen van een rekening voor een client voor het gebruik van de gemeten dienst. Als er sprake is van een rekening die op basis van gebruik wordt vastgesteld (usage−sensitive billing) is het belangrijk om betrouwbare gegevens over het gebruik te hebben. Er moet ook tegenover de client te verantwoorden zijn dat de accounting gegevens waarop de rekening is gebaseerd betrouwbaar zijn. Soms zijn er real−time billinggegevens nodig als er sprake is van een beperkt krediet. De accounting gegevens zelf bevatten echter geen prijsgegevens, omdat deze meestal niet bij het service element beschikbaar zijn. Billinggegevens zijn vaak afhankelijk van gegevens van de client en kunnen bijvoorbeeld met speciale kortingen of piek− en daltarieven werken. Omdat accounting gegevens zich direct laten vertalen in opbrengsten stelt billing strenge eisen aan accounting met betrekking tot betrouwbaarheid, beveiliging en fraudegevoeligheid.

Cost allocation

Bij cost allocation worden accounting gegevens gebruikt om gemaakte kosten te verdelen over projecten, afdelingen of zakenpartners die een gemeenschappelijke resource gebruiken. Hierbij kunnen bijvoorbeeld de kosten voor telefoonverkeer worden verdeeld op basis van gemeten gebruik. Bij cost allocation is er, net als bij billing, een financieel belang, wat er voor zorgt dat er eisen moeten worden gesteld aan de fraudegevoeligheid. Cost allocation onderscheidt zich van billing, doordat hier geen sprake is van opbrengsten van het leveren van een dienst, maar van het verdelen van kosten van gebruikte diensten. Usage−sensitive cost allocation kan voor een efficiënter gebruik van middelen leiden doordat kosten reëel worden teruggekoppeld.

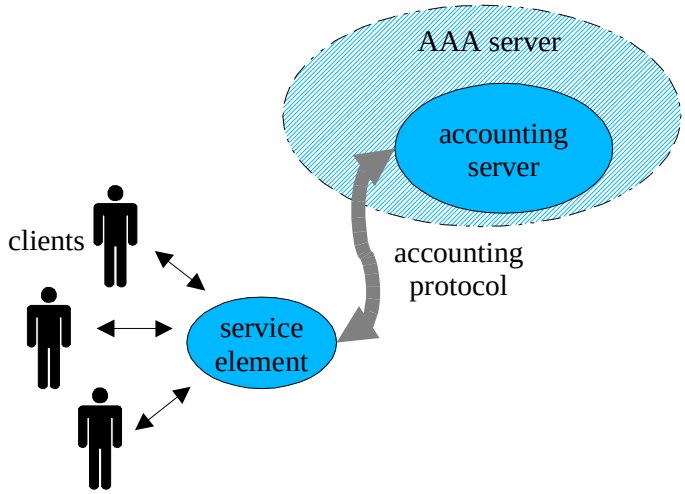
2.1 Accounting protocol

Accounting gegevens worden met behulp van een accounting protocol door het service element aan een accounting server doorgegeven. De accounting server kan dan trendanalyse, auditing, billing of cost allocation uitvoeren of dit aan een gespecialiseerde server overlaten. Bij accounting wordt dus nog geen waarde aan sessies gekoppeld, dit gebeurt pas als billing wordt uitgevoerd.

De accounting server is vaak onderdeel van een AAA server die ook authenticatie en autorisatie uitvoert. Het accounting protocol is daarom ook vaak direct onderdeel van een geheel AAA protocol. Een accounting protocol bestaat uit

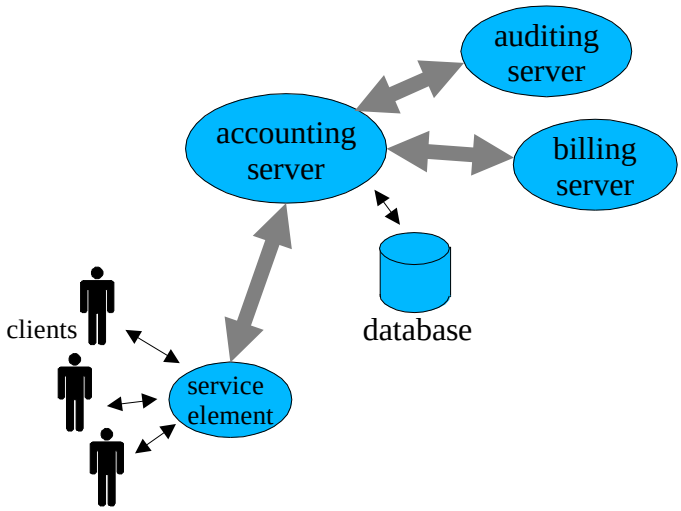
- een transportprotocol,
- een verzameling mogelijke berichten en
- een record format.

Een transport protocol draagt zorg voor het vervoer van accounting berichten over een netwerk. De verzameling berichten definieert de verschillende soorten accounting berichten die kunnen worden verstuurd. Een record format bepaalt op welke manier de accounting gegevens worden vastgelegd.



Figuur 2.1: service element en accounting server.

Accounting berichten zijn berichten die tussen het service element en de accounting server worden uitgewisseld. De accounting server heeft een verzamelende taak. Service elements genereren in het algemeen bij begin en eind van het aanbieden van de dienst accounting berichten, maar ook tijdens het aanbieden van de dienst kunnen zogenaamde interim berichten worden verstuurd. Accounting servers kunnen start−, stop− en interim−berichten die tot dezelfde sessie behoren gebruiken om session records te maken. Zo kunnen bepaalde accounting attributen worden gesommeerd en kunnen ook eventuele dubbele berichten worden uitgefilterd. Zelfs bij een crash van het service element kan, als er voldoende interim−berichten zijn verstuurd, een relatief betrouwbare session record worden gegenereerd.



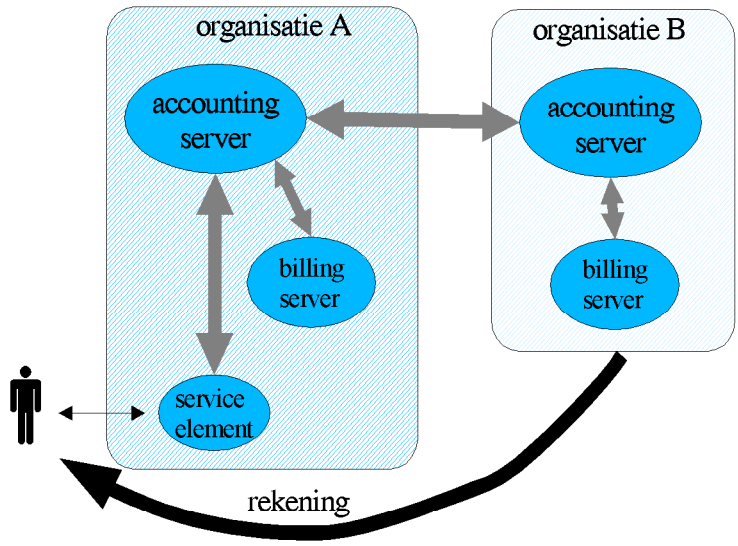
Figuur 2.2: voorbeeld van een accounting server en andere servers.

Aan de hand van de session records kan trendanalyse, auditing, billing en cost allocation worden uitgevoerd. Hiermee wordt inzicht gegeven in het gebruik. Dit is ook voor de gebruiker van de dienst van belang. Die kan bepalen in welke mate veranderingen in zijn structuur van invloed zijn op het gebruik van de dienst. Zo kan hij de kosten terugbrengen. De session records kunnen dan verder worden opgeslagen of getransporteerd. Session records worden bijvoorbeeld door een billing server gebruikt om rekeningen op te stellen. Bij het transport van accounting server naar andere servers kan ook het accounting protocol gebruikt worden. Ook is het mogelijk om hier een apart protocol voor te gebruiken.

Het kan ook nodig zijn dat, voordat het session record compleet is, accounting gegevens aan een opvolgende server worden doorgespeeld. Als een auditing server bijvoorbeeld misbruik vermoedt en tijdens de sessie wil kunnen ingrijpen, dan moeten gegevens van het service element zo snel mogelijk bij de auditing server terecht komen.

2.2 Inter-domain accounting

Bij het aanbieden van diensten is het voor de aanbieder vaak belangrijk om accounting toe te passen. Als er echter nog een organisatie belang heeft bij de dienst kan het voorkomen dat deze ook inzicht wil hebben in de accounting gegevens. Dit gebeurt bijvoorbeeld als een client diensten afneemt van organisatie A (zie Figuur 2.3), terwijl hij een zakelijke relatie met organisatie B heeft. De client krijgt dus uiteindelijk een rekening van organisatie B. Zo lijkt het voor de client alsof hij diensten afneemt van organisatie B. In dit voorbeeld wordt organisatie A het local domain en organisatie B het home domain genoemd.

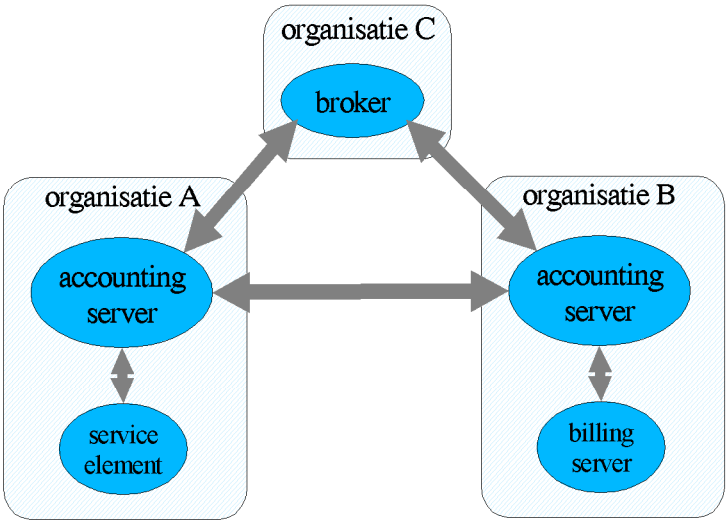


Figuur 2.3: voorbeeld van inter-domain accounting.

Bij inter-domain accounting worden accounting gegevens tussen verschillende organisaties uitgewisseld. Bij intra-domain accounting blijven de accounting gegevens binnen dezelfde organisatie. De accounting gegevens worden dan via een lokale accounting server in het local domain doorgespeeld naar de accounting server in het home domain waarvoor de accounting gegevens daadwerkelijk zijn bestemd. De lokale accounting server werkt dan als proxy [rfc2194], [rfc2607]. Het kan nodig zijn dat de lokale accounting server extra informatie, zoals extra kosten voor het inter-domain gebruik, aan de accounting gegevens toevoegt. Ook blijven de accounting gegevens interessant voor lokale doeleinden, omdat bijvoorbeeld een rekening aan organisatie B moet worden gestuurd. Zo kan de uiteindelijke communicatie via verschillende proxies lopen.

Omdat accounting gegevens mogelijk over een open netwerk gaan of door meerdere proxies worden doorgegeven, stelt dit extra eisen aan de beveiliging. Het moet niet zo zijn dat een accounting proxy kan frauderen door gegevens te veranderen. Om de communicatie tussen de twee organisaties te regelen wordt vaak gebruik gemaakt van een broker.

Deze broker (zie Figuur 2.4), die bij een trusted third party behoort, bemiddelt in de communicatie tussen de twee accounting servers. Dit kan gebeuren door de gehele communicatie via de broker te laten verlopen of door de broker te laten bemiddelen in het opzetten van een beveiligd kanaal tussen de twee accounting servers. De broker wordt dan, als onafhankelijke partij, in de bemiddeling gebruikt.



Figuur 2.4: gebruik van een broker.

2.3 Overhead

Het genereren, transporteren en verwerken van accounting gegevens behoort niet tot de dienstverlening zelf. Voor accounting worden echter ook resources gebruikt. Deze resources bestaan voornamelijk uit gebruik van bandbreedte op het netwerk, het gebruik van processortijd op service elements en accounting servers en het gebruik van opslagruimte voor (tijdelijke) opslag van accounting gegevens [ietf−aaa−acct, 6.2].

Voor het transporteren van accounting gegevens worden verbindingen op het netwerk tot stand gebracht. De hoeveelheid benodigde bandbreedte is evenredig met de hoeveelheid getransporteerde accounting gegevens vermeerderd met de overhead voor transport. Omdat enkele accounting berichten over het algemeen erg klein zijn loopt de overhead voor transport, als accounting berichten individueel worden verzonden, aardig op. Het transporteren gebruikt niet alleen netwerkbandbreedte, maar er zijn voor een verbinding ook buffers en handles nodig. Het uitvoeren van batching en het gebruik van compressie kunnen de hoeveelheid gebruikte netwerkbandbreedte verminderen.

Het opslaan van accounting gegevens die wachten op transport kost een hoeveelheid geheugenruimte op het service element. Deze ruimte hangt af van de transportmethode. Ook op de accounting server is geheugenruimte nodig om ontvangen gegevens op te slaan. Ook als de accounting server de gegevens meteen doorstuurt is er een bufferruimte nodig om gegevens te bewaren totdat ze zijn getransporteerd. Voor de uiteindelijke opslag van accounting gegevens in een database of logfile is een grote hoeveelheid geheugenruimte nodig. Vaak moeten accounting gegevens voor langere perioden worden bewaard. Het gebruik van compressie bij opslag van accounting gegevens kan hierin een aanzienlijke besparing opleveren. Ook het in een vroeg stadium elimineren van dubbele gegevens (bijvoorbeeld bij elkaar horende interim berichten) kan een aardige besparing opleveren.

Het genereren en verwerken van accounting gegevens op service elements, accounting servers en proxies kost een zekere hoeveelheid processortijd. Als compressie van accounting gegevens wordt gebruikt kost dit eveneens processortijd. Om start−, stop− en interimberichten bij elkaar te zoeken is een hoeveelheid administratie van sessies op de accounting server nodig. Accounting gegevens moeten vaak worden beschermd tegen misbruik. Hiervoor worden beveiligingstechnieken toegepast. Het coderen van gegevens kost over het algemeen een aanzienlijke hoeveelheid resources. Tevens is het vaak nodig om sleutels tussen de verschillende partijen te distribueren.

Bij het gebruik van accounting moet rekening worden gehouden met de overhead die accounting met zich meebrengt. De hoeveelheid overhead hangt in het algemeen af van de detaillering en de betrouwbaarheid van de verslaglegging. Bij service elements die snel veel diensten moeten kunnen leveren, zoals routers, kan het berekenen en vastleggen van detailinformatie aanzienlijke rekenkracht en netwerkbandbreedte vergen.

Bij het gebruik van accounting moet dus een afweging worden gemaakt tussen de opbrengsten van accounting (billing, auditing, etc.) en de kosten ervan (overhead). Bij deze afweging moet worden bepaald tot op welk niveau detailgegevens moeten worden bijgehouden. Het is mogelijk om, na inspectie, bepaalde detailgegevens te sommeren en slechts een beperkt aantal gegevens te bewaren.

2.4 Push vs Pull

Het uitvoeren van accounting kan in principe op twee manieren gebeuren. De eerste mogelijkheid is dat bij het optreden van een event accounting gegevens vanuit het service element aan de accounting server worden doorgegeven. De tweede is dat een accounting server met enige regelmaat alle service elements vraagt of er nog accounting gegevens te versturen zijn.

In het eerste geval is er sprake van push van accounting gegevens. Dit wordt ook wel event−driven accounting genoemd. In dit geval zijn accounting gegevens snel op de plaats van bestemming omdat niet gewacht hoeft te worden totdat de accounting server vraagt of er gegevens beschikbaar zijn. Als er sprake moet zijn van real−time accounting is dit in veel gevallen de beste oplossing. Op het service element is weinig bufferruimte nodig, omdat accounting gegevens maar erg kort (totdat er een bevestiging is terug gekomen) hoeven te worden bewaard. Deze aanpak leent zich minder goed voor batching, maar het is mogelijk om batching uit te voeren door te wachten tot een buffer voor een bepaalde hoeveelheid is gevuld.

In het tweede geval is er sprake van pull van accounting gegevens. De accounting server gaat alle service elements af om te vragen of er accounting gegevens beschikbaar zijn. Dit wordt ook wel polling genoemd. Het voordeel van deze aanpak is dat er geen verbinding per event wordt opgezet, maar een verbinding per service element. Ook is het mogelijk om polling zo in te delen dat het netwerk niet in één keer wordt overbelast met allemaal accounting berichten. Ook leent deze aanpak zich beter voor batching, waarmee de efficiëntie van de transport kan worden verhoogd. Een nadeel van deze aanpak is dat er op het service element een aanzienlijke hoeveelheid bufferruimte nodig is om de accounting gegevens op te slaan. Verder is het toepassen van polling in een netwerk met veel service elements die niet allemaal regelmatig accounting gegevens produceren inefficiënt. Zo is deze oplossing niet toe te passen bij inter−domain accounting.

Het is ook mogelijk om een een hybride oplossing te kiezen: event−driven polling. Hierbij stuurt een service element één keer een berichtje aan een accounting server dat het accounting gegevens beschikbaar heeft, waarna de accounting server deze gegevens, op zijn eigen tempo, ophaalt. Deze aanpak heeft de voordelen van polling waarbij scheduling van accounting transport kan worden toegepast, terwijl alleen die service elements die accounting gegevens beschikbaar hebben gepolled hoeven te worden. Het is efficiënter dan event−driven accounting, omdat het service element maar één keer een bericht aan de accounting server hoeft te sturen. Event−driven polling is in dit geval wel mogelijk in een inter−domain accounting omgeving. Voor real−time accounting blijft echter een event−driven aanpak nodig zijn.

3. Toepassingen

In dit hoofdstuk zullen enkele toepassingen van accounting worden behandeld, met als doel het inzicht geven in de soorten diensten waarbij accounting wordt toegepast en op welke manier dat gebeurt. Niet voor alle toepassingen zijn trendanalyse, auditing, billing en cost allocation mogelijk of vanzelfsprekend.

Veel van de hier behandelde toepassingen stellen verschillende eisen aan accounting. Vaak is er sprake van een financieel belang, wat er voor zorgt dat accounting zeer zorgvuldig moet gebeuren.

3.1 Inbellen

Internet Service Providers (ISP's), ook wel Internet Access Provides (IAP's) genoemd, bieden inbelfaciliteiten aan. Hierdoor kan via de telefoon of kabel met bijvoorbeeld PPP een verbinding met internet worden gemaakt. Voor het leveren van inbelfaciliteiten is een overvloed aan producten (Network Access Servers) te koop. De meeste van deze producten ondersteunen het RADIUS (Remote Authentication Dial In User Service) protocol [rfc2138]. Daarnaast worden vaak diverse andere protocollen ondersteund. Bij inbellen zijn authenticatie en autorisatie van groot belang. De meeste van de protocollen voor Netwrok Access Servers zijn dan ook authenticatie en autorisatie protocollen waaraan later een accounting deel is toegevoegd.

Bij inbellen is accounting zeer belangrijk. Bij een aantal ISP's wordt usage−sensitive billing toegepast, maar ook bij de ISP's die met flat−rate billing werken is het belangrijk trendanalyse en auditing uit te voeren.

3.2 Telefonie

In de telefonie wordt accounting al erg lang toegepast. De rekening die een telecomprovider maakt is gebaseerd op specifiek gebruik. Vroeger werden gegevens over gevoerde telefoongesprekken via papier en later via magneetbanden vervoerd, waarna het werd verwerkt.

In de telefonie worden vaak ingewikkelde billing strategieën gebruikt, met bijvoorbeeld een aantal door de klant gekozen nummers waarmee tegen een gereduceerd tarief kan worden gebeld. Ook is het mogelijk om een zeer gespecificeerd verslag te maken met een overzicht van welke nummers wanneer en hoe lang zijn gebeld. Voor dit alles is een zeer betrouwbaar accounting systeem nodig.

Met mobiele telefonie wordt het nog ingewikkelder, omdat de client kan bewegen en van verschillende service elements gebruik kan maken om een enkele sessie te voeren. Dit gebeurt als een client van een cel naar een andere cel beweegt.

3.3 Telefooncentrale

Veel bedrijven hebben intern een PBX (Private Branch Exchange). Deze kan worden gebruikt voor intern telefoonverkeer, maar ook voor telefoonverkeer naar buiten. Om na te gaan of de buitenlijn niet wordt misbruikt, om kosten van gebruik van de buitenlijn op de juiste afdeling of op het juiste project te krijgen of om de rekening van de telecomaanbieder te controleren [billaudit] kan hier accounting worden toegepast.

Veel telefooncentrales kunnen Call Detail Records genereren en die met SMDR (Station Message Detail Recording) transporteren. Deze gegevens kunnen dan met een pc worden gebruikt om rapporten te genereren.

3.4 Interne netwerkelementen

Van interne netwerkelementen zoals routers, firewalls etc. is het vaak belangrijk inzicht te krijgen in het gebruik ervan [rfc1272]. Het gaat hier echter vaak om een grote hoeveelheid sessies die uit een enkel event bestaan. Vaak is het onmogelijk om hier erg gedetailleerde accounting toe te passen, omdat dan voor elk binnenkomend bericht een accounting bericht zou moeten worden gegenereerd. Dit betekent bijvoorbeeld bij routers een verdubbeling van het netwerkverkeer. Het wordt nog ingewikkelder als deze accounting berichten op zich weer via een router op hun bestemming komen. Bij routers is het wel belangrijk om een globaal inzicht te hebben in het netwerkverkeer, dit met het oog op trendanalyses voor de netwerkstructuur.

Bij firewalls wordt vaak van een beperkte hoeveelheid transacties relatief uitvoerig bericht. Omdat firewalls voor beveiliging van een netwerk zorgen is auditing van dit verkeer natuurlijk belangrijk. Hetzelfde geldt voor border−routers die twee netwerken van verschillende organisaties met elkaar verbinden.

3.5 Website hosting

Website hosting is het aanbieden van schijfruimte op een computer die verbonden is met het internet voor het plaatsen van internet pagina's. Bedrijven die zelf geen snelle internetverbinding hebben of het beheer van apparatuur buiten de eigen organisatie willen houden kunnen gebruik maken van deze diensten. Hierbij kan ook worden gedacht aan zaken als het beheren van databases voor de website of andere dienstverlening die daarmee samenhangt.

Aan hosting en andere soortgelijke diensten hangt natuurlijk een prijskaartje. Van deze diensten is het dus wenselijk inzicht te hebben in het gebruik ervan. Bij hosting wordt vaak een afgesproken hoeveelheid schijfruimte verhuurd en gereserveerd. Het werkelijk gebruik is vaak interessant om analyses te kunnen maken voor de totaal beschikbare ruimte. Hiermee kan worden bekeken of aan de voorwaarden voor het gebruik wordt voldaan.

Bij hosting is het vaak ook belangrijk inzicht te krijgen in de frequentie waarmee de pagina's worden opgevraagd. Door logfiles van de webserver te bekijken kan inzicht worden verkregen in de netwerkbelasting en de frequentie waarmee de gegevens worden opgevraagd.

Deze dienstverlening vereist een andere vorm van accounting dan bijvoorbeeld bij inbellen wordt gebruikt. Er is geen sprake van korte sessies waarover wordt afgerekend. De sessies duren in het algemeen lang en hoeven niet te eindigen. Bij hosting en soortgelijke dienstverlening is het dus nodig om op basis van delen van de sessie accounting uit te voeren. Interessant is de mogelijkheid om met een accounting systeem gegevens over het gebruik van schijfruimte en het gebruik van netwerkbandbreedte te combineren.

3.6 Application hosting

Application hosting is vergelijkbaar met website hosting met het verschil dat er een applicatie wordt verhuurd. Application Service Providers (ASP's) verzorgen de apparatuur en de software voor een bepaalde klant. De klant gebruikt deze via een netwerk. De klant hoeft zich niet bezig te houden met het beheer en onderhoud aan de apparatuur en software. Soms kunnen applicaties door meerdere klanten worden gebruikt wat de kosten drukt.

Application hosting lijkt daarmee een klein beetje op het gebruik van rekencentra met mainframes, waarbij processortijd op een mainframe kan worden gekocht om verschillende programma's te draaien. Net als bij mainframes is bij application hosting accounting van belang. Het is belangrijk om inzicht te hebben in de resources die worden gebruikt om een realistische rekening te kunnen opmaken.

3.7 Roaming

Roaming [rfc2194] is het gebruik van diensten bij een andere provider dan waarmee een overeenkomst bestaat. Dit kan bijvoorbeeld gebeuren bij inbellen waarbij een client is aangesloten bij een internet provider (home ISP) en voor het inbellen gebruik maakt van de diensten van een lokale internet provider (local ISP). Zo kunnen verschillende internet providers samenwerken om gezamenlijk een groter gebied te bestrijken. Deze afspraak wordt een roaming agreement genoemd. Bij roaming is sprake van inter−domain accounting.

Bij roaming worden de gegevens over de gebruiker in het algmeen bij de home ISP opgeslagen. Bij het inloggen (bij de local ISP) wordt de authenticatie en autorisatie mede door de home ISP geregeld. Accounting gegevens zijn voor zowel home ISP als local ISP belangrijk. Voor de home ISP worden accounting gegevens gebruikt om klanten rekeningen te kunnen sturen en voor de local ISP worden de accounting gegevens gebruikt om de home ISP een rekening te kunnen sturen.

Bij roaming worden accounting gegevens over grote afstand getransporteerd. Omdat hierbij financiële belangen een rol spelen vereist deze vorm van accounting een grote betrouwbaarheid. Accounting gegevens moeten bijvoorbeeld niet door de local ISP kunnen worden vervalst. Omdat de gegevens door de local ISP worden gegenereerd moet de home ISP de uiteindelijke gebruiker kunnen autentificeren en er zeker van zijn dat de accounting gegevens echt zijn. De local ISP moet aan de andere kant de zekerheid hebben dat de home ISP de kosten voor de verbinding accepteert. Beide ISPs moeten later niet kunnen ontkennen dat de accounting gegevens daadwerkelijk zijn verstuurd.

Omdat het niet haalbaar is alle ISP's met alle andere ISP's roaming agreements te laten maken, kunnen brokers worden gebruikt waarmee een roaming consortium kan worden gevormd. De brokers handelen dan als onafhankelijk tussenpersoon en kunnen zaken als beveiliging regelen.

3.8 Resource reservation

Het RSVP protocol kan worden gebruikt om Quality of Service (QoS) te leveren op internet. Internet is een best−effort netwerk waarbij geen garanties kunnen worden gegeven over de beschikbare bandbreedte. Met RSVP kan een hoeveelheid bandbreedte of nodige respons van het netwerk worden gereserveerd. Multimedia applicaties kunnen dit bijvoorbeeld nodig hebben. RSVP is gedefinieerd in [rfc2205].

Bij RSVP wordt vanuit de ontvanger een verzoek gedaan om bandbreedte te reserveren vanaf een bepaalde zender [rsvpwww]. Onderweg worden verschillende reserveringen samengevoegd. RSVP gebruikt de aanwezige routing in internet en verzorgt zelf geen transport. Het zorgt alleen voor de reservering. Het is ontworpen om multicast verbindingen te ondersteunen, zoals radio uitzendingen en dergelijke.

Bij het bieden van een gegarandeerde bandbreedte komt accounting vanzelfsprekend om de hoek kijken. Diensten met een gegarandeerde kwaliteit hebben over het algemeen een prijskaartje. In [rsvpacc] wordt een opzet voor het gebruik van accounting in RSVP beschreven. Hierin wordt echter geen accounting protocol gepresenteerd waarmee dit te verwezenlijken is. De beschreven methode is gebaseerd op eenvoudige afspraken tussen netwerk providers waarbij verkeer tussen de twee partijen wordt verrekend, zoals dat ook bij best effort verkeer gebruikelijk is. Gebruikers worden dan afgerekend op een deel best effort verkeer en een deel gereserveerd verkeer over het netwerk. Providers krijgen op hun beurt weer rekeningen in dezelfde vorm van de ISP waar het verkeer in eerste instantie naar toe gaat.

3.9 Overig

Verder zijn er nog een groot aantal gebieden waar accounting wordt of kan worden toegepast. Bij het gebruik van processortijd op een mainframe, printen via een centrale printerserver, gebruik van een fileserver waarbij een stuk schijfruimte wordt verhuurd, teleconferencing, pay−television, logging en nog vele andere zaken kan accounting worden gebruikt.

Bij teleconferencing is het denkbaar dat er via verschillende media door verschillende personen en organisaties een conference tot stand wordt gebracht. Deze verschillende media produceren naar alle waarschijnlijkheid voor alle aangesloten gebruikers (met verschillende geografische locaties) accounting gegevens. Het is wenselijk om al deze gegevens te kunnen groeperen en daar een rekening voor te sturen.

Ook zou accounting wellicht gebruikt kunnen worden om structuur te brengen in logging bij webservers, ftp servers en andere soortgelijke diensten. Deze servers produceren grote logfiles met erg veel informatie. Om deze informatie op een eenvoudige manier te kunnen verwerken en te transporteren voor bijvoorbeeld remote auditing zou een accounting protocol kunnen worden gebruikt. Ook zouden logfiles van verschillende servers centraal kunnen worden bijgehouden.

Telefoon en fax via IP zijn in de groei, net als zaken als business email, application hosting en andere diensten die ISP’s aanbieden. Ook bij deze zaken kan accounting een grote rol spelen.

4. Requirements

In dit hoofdstuk zullen diverse requirements voor accounting en accounting protocollen die algemeen toepasbaar zijn worden behandeld. Een accounting protocol valt in het algemeen onder te verdelen in

- een transport protocol,
- een verzameling berichten en
- een record format.

Een transport protocol draagt zorg voor het vervoer van accounting berichten over een netwerk. De verzameling berichten definieert de verschillende soorten accounting berichten die kunnen worden verstuurd. Een record format bepaalt op welke manier de accounting gegevens worden vastgelegd. Het is aan te bevelen om de requirements van deze onderdelen zo veel mogelijk gescheiden te houden, zodat deze eventueel afzonderlijk kunnen worden ontworpen.

Eerst worden algemene requirements, die aan accounting en accounting protocollen worden gesteld behandeld. Hierna komen de security requirements aan bod. Vervolgens zullen de benodigde berichten voor accounting worden behandeld in de accounting event requirements. Tenslotte worden requirements voor het transportprotocol en het record format behandeld.

De requirements zijn vaak onderling afhankelijk en kunnen dan ook niet los van elkaar worden gezien. Veel requirements uit het algemene gedeelte komen bijvoorbeeld terug in de overige requirements.

De requirements in dit document zijn van toepassing op een algemene manier van accounting en accounting protocollen welke voor meerdere diensten toepasbaar zijn. Deze requirements zijn afkomstig uit de verschillende vakgebieden die zich met accounting bezig houden. De requirements die hier worden behandeld, zijn vrijwel allemaal opgesomd in [arkko−acctrqlis], [ietf−aaa−na−reqts, 4.3] en [ietf−nasreq−criteria, 8.4]. Bij de requirements zelf worden referenties genoemd waar meer details over de requirements en de herkomst zijn te vinden.

Omdat accounting vaak wordt toegepast samen met een authenticatie en autorisatie protocol en omdat veel requirements uit authenticatie en autorisatie ook bij accounting terugkomen, is het handig om voor deze protocollen een gelijksoortige opzet te kiezen. Hierdoor hoeven niet alle drie de protocollen volledig afzonderlijk van elkaar te worden ontworpen. Zo zal het vaak voorkomen dat hetzelfde transport protocol en een gezamenlijk record format worden gebruikt. Dit maakt het ontwerp van service elements eenvoudiger. In dit hoofdstuk worden echter alleen de specifieke eisen voor een accounting protocol behandeld.

4.1 Algemene requirements

Dit zijn requirements die aan het gehele accounting proces worden gesteld en betrekking hebben op alle aspecten van accounting. Deze requirements zullen in de requirements van de specifieke delen vaak terugkomen.

4.1.1 Real−time accounting (MUST)

Real−time accounting is het verwerken van accounting informatie binnen een bepaald tijdsbestek. Deze tijdslimiet wordt in het algemeen opgelegd om financiële risico's te vermijden of om real−time auditing uit te voeren. Een snelle terugkoppeling van gegevens over gebruik kan bijvoorbeeld nodig zijn om, bij overschrijding van een kredietlimiet, via het autorisatie gedeelte de dienstverlening snel te staken. Hoewel accounting in principe alleen voorziet in de verslaglegging van de diensten die een service element biedt, kunnen deze gegevens ook worden gebruikt om inzicht te krijgen in de huidige toestand van het service element.

Real−time accounting eist dat het transport protocol de berichten snel kan versturen, dat de accounting server snel accounting gegevens aan een billing of auditing server doorstuurt en dat de billing of auditing server deze gegevens snel verwerkt en hier eventueel actie op onderneemt.

Na het optreden van het event moeten de accounting gegevens dus binnen een bepaalde tijd kunnen zijn verwerkt. De tijdslimiet is afhankelijk van de toepassing, maar zou bijvoorbeeld een seconde kunnen zijn.

Referenties: [ietf−aaa−acct, 5.1, 5.6], [ietf−nasreq−criteria, 8.4.1.2], [arkko−acctreq, 6.1], [rfc2477, 4.3] en [ietf−mobileip−aaa−reqs, 3.1].

4.1.2 Archival accounting (MUST)

Bij archival accounting gaat het om het verzamelen van alle accounting gegevens, het in geval van gegevensverlies missende accounting gegevens zo goed mogelijk te reconstrueren en het archiveren van accounting gegevens voor een bepaalde tijd. Juridische en financiële behoeften kunnen archival accounting voorschrijven en kunnen opleggen dat accounting gegevens vertrouwelijk worden behandeld.

Bij usage−sensitive billing betekent het verlies van accounting gegevens in het algemeen verlies van inkomsten. Dit betekent dat het billingproces, en daarmee het accounting proces aan dezelfde standaarden moeten voldoen als de financiële verslaglegging.

Archival accounting legt één van de striktste eisen aan accounting met betrekking tot vertrouwelijkheid, juistheid en volledigheid van accounting gegevens. Dit betekent dat ook bij uitval van service elements, accounting servers, netwerk of andere onderdelen van het accounting proces, geen of zo min mogelijk accounting gegevens verloren gaan.

Bij archival accounting is het vooral belangrijk dat, op alle elementen die met de accounting gegevens in aanraking komen, de accounting gegevens niet verloren gaan bij het uitvallen van het element of het netwerk. Dit betekend dat er op deze elementen non−volatile storage aanwezig moet zijn. Archival accounting stelt dan ook eerder een eis aan het gehele accounting systeem dan aan het protocol. In het protocol moet het echter wel mogelijk zijn om deze middelen te gebruiken.

Referentie: [ietf−aaa−acct, 5.6].

4.1.3 Batch accounting (MUST)

Soms is het nodig om een grote hoeveelheid accounting gegevens in één keer te versturen. Dit kan zo zijn als accounting gegevens voor een off−line audit worden getransporteerd, maar ook als het service element accounting gegevens buffert. Redenen voor buffering zijn het niet bereikbaar zijn van de accounting server of een efficiënter gebruik van bandbreedte. Met gebufferde gegevens kan een betere compressie worden behaald en buffering veroorzaakt minder overhead.

Het protocol moet dus het versturen van gebufferde accounting gegevens in batches ondersteunen. Dit betekent onder andere dat het transportprotocol het versturen van grote hoeveelheden gegevens moet ondersteunen en dat servers deze hoeveelheid gegevens aan moeten kunnen.

Referenties: [ietf−aaa−acct, 6.1.6], [ietf−nasreq−criteria, 8.4.1.3] en [arkko−acctreq, 6.1].

4.1.4 Minimale overhead (SHOULD)

Het genereren, transporteren en verwerken van accounting gegevens behoort niet tot de dienstverlening zelf. De overhead bestaat uit kosten voor het gebruik van bandbreedte, opslagruimte en processortijd. Om deze overhead zo minimaal mogelijk te houden is het belangrijk een zo eenvoudig en compact mogelijk accounting protocol te gebruiken.

Het record formaat voor transport en opslag zou zo klein mogelijk moeten zijn. Voor versturen en opslag kan compressie worden gebruikt. Vooral bij het versturen in batches kan dit een aanzienlijke besparing opleveren. Compressie kost echter processortijd en er moet dus een afweging worden gemaakt tussen de verschillende kosten.

In het algemeen moet er een afweging worden gemaakt tussen de verschillende requirements. Het protocol moet zo zijn ingericht dat er een minimale overhead ontstaat, zonder dat daardoor andere requirements tekort worden gedaan. De detaillering van de verslaglegging zal in het algemeen direct samenhangen met de hoeveelheid overhead.

Referentie: [ietf−aaa−acct, 6.2].

4.1.5 Schaalbaar (MUST)

Schaalbaarheid (scalability) is de mogelijkheid van een product om goed te functioneren bij een verandering in omvang van het gebruik of omvang van de context. Dit betekent dat bij een toename van het gebruik, het product niet meer dan een evenredige hoeveelheid resources extra nodig heeft. Met de huidige groei van het internet is het belangrijk om accounting systemen schaalbaar te ontwerpen.

Dit betekent bijvoorbeeld dat, bij een verdubbeling van de hoeveelheid gegenereerde accounting events niet veel meer dan een verdubbeling van de grootte van de accounting server nodig is. Het moet mogelijk zijn accounting zo uit te voeren dat het aantal verbindingen gelijk is aan het aantal service elements die informatie voor een accounting server beschikbaar hebben als er geen sprake van real−time accounting hoeft te zijn. Zo groeit het aantal verbindingen niet met de hoeveelheid aangeboden accounting requests, maar met de hoeveelheid aangesloten service elements. Bij het uitvoeren van accounting is het nodig om startberichten bij bijbehorende stop− en interimberichten te zoeken. Accounting servers moeten dus per actieve sessie informatie bijhouden. Dit betekent dat er een limiet zit aan het aantal bij te houden sessies.

Referenties: [ietf−aaa−acct, 6.3], [arkko−acctreq, 6.2], [wang−aaa−cel−req, 7], [rfc2477, 4.2.3] en [ekstein−nasreq−protocomp, 2.6].

4.1.6 Ondersteuning van eindige sessies (MUST)

Eindige sessies hebben een tijdstip waarop ze beginnen en een later tijdstip waarop ze eindigen. Dit is het meest algemene geval waarbij accounting wordt toegepast. Het bieden van een inbelverbinding vormt bijvoorbeeld een sessie. Het accounting protocol moet accounting van eindige sessies kunnen ondersteunen.

Referentie: [arkko−acctreq, 6.5].

4.1.7 Ondersteuning van oneindige sessies (MUST)

Bij oneindige sessies ligt het einde van de sessie niet in de nabije toekomst of hoeft helemaal niet op te treden. Voorbeelden van dit soort sessies zijn het aanbieden van web−space. Van deze diensten kan in het algemeen niet na het aflopen van de dienst een session record worden gegenereerd. Dit betekent dat billing op basis van alleen session records bij dit soort dienstverlening tekort schiet.

Het accounting protocol moet bij dit soort sessies accounting kunnen toepassen. Dit kan bijvoorbeeld door van een deel van de sessie een soort session record te maken. Zo kan de oneindige sessie in meerdere aansluitende sessies worden opgedeeld.

Referentie: [arkko−acctreq, 6.5].

4.1.8 Ondersteuning van ondeelbare events (MUST)

Bij ondeelbare events is er geen sprake van een sessie, maar van een enkele gebeurtenis. Voorbeelden hiervan zijn het opvragen van een web−pagina. Het is mogelijk om voor zulke events een apart start en stop bericht te versturen, maar dit is verspilling van resources en is niet bepaald een mooie oplossing. Mooier is om hiervoor een enkel bericht te kunnen versturen. Het accounting protocol moet dergelijke events ondersteunen.

Referentie: [arkko−acctreq, 6.5].

4.1.9 Inter−domain accounting (MUST)

Het moet mogelijk zijn om het accounting protocol te gebruiken in een omgeving waar sprake is van inter−domain accounting. Hierbij moet ervan worden uitgegaan dat accounting gegevens over een grote afstand en over een open netwerk worden vervoerd.

Accounting servers van verschillende organisaties met eventueel verschillende versies van hetzelfde protocol moeten met elkaar kunnen communiceren en gegevens moeten, afhankelijk van de toepassing, afdoende kunnen worden beveiligd.

4.1.10 Meerdere accounting servers (MUST)

Meerdere accounting servers worden gebruikt om, bij het uitvallen van een accounting server, accounting taken over te nemen. Ook kan de werklast over verschillende servers worden verdeeld (load balancing). Verder is het mogelijk dat accounting gegevens voor verschillende accounting servers interessant zijn.

In het accounting protocol moet het mogelijk zijn om met meerdere accounting servers te werken. Het moet mogelijk zijn om accounting berichten naar verschillende accounting servers te sturen. Dit laatste kan eventueel worden overgelaten aan een speciale accounting proxy of broker. Het accounting protocol moet ook kunnen worden toegepast in een model waarbij geen sprake is van een eenvoudige client−server omgeving.

Bij het sturen van accounting berichten naar meerdere servers kan het gebeuren dat accounting berichten dubbel worden opgenomen. Het accounting protocol moet voorzien in een methode om dubbele berichten over verschillende servers uit te filteren. Ook moeten berichten die tot dezelfde sessie behoren en over verschillende servers zijn verdeeld gezamenlijk kunnen worden verwerkt.

Referenties: [ietf−aaa−acct, 5.2, 6.1.4, 6.1.5] en [ietf−nasreq−criteria, 8.1.2.4].

4.1.11 Samengestelde diensten (SHOULD)

Het komt voor dat verschillende service elements verschillende diensten leveren die bij een enkele dienst van een hoger niveau hoort. Hierbij kan worden gedacht aan teleconferencing waarbij alle verbindingen die bij de teleconference worden gebruikt bij één en dezelfde dienst horen. Ook kan het gebeuren dat verschillende service elements gegevens over een enkele sessie produceren als bijvoorbeeld de client, tijdens het afnemen van de dienst, van service element verandert. Dit gebeurt bijvoorbeeld bij mobiele telefonie met bewegende gebruikers.

In beide gevallen worden vanuit verschillende service elements accounting gegevens geproduceerd die bij elkaar horen. Het accounting protocol moet in een mechanisme voorzien waarbij gegevens uit deze verschillende sessies kunnen worden samengenomen.

Hierbij is het echter vaak van belang geen informatie te verliezen over de afzonderlijke onderdelen van de sessies. Dit betekent dat het eenvoudig sommeren van de gegevens geen goede optie is. Het moet mogelijk zijn om sessies te definiëren die worden gevormd door verschillende andere sessies.

Referenties: [blount−acct−service, 4.4.3] en [wang−aaa−cel−req, 7].

4.2 Security requirements

Security requirements hebben betrekking op de beveiliging van de accounting gegevens. Deze requirements komen vooral voort uit de financiële en juridische eisen aan accounting. Dit betekent dat er bij het transporteren van accounting gegevens onderweg geen gegevens vervalst kunnen worden of door derden te lezen zijn. De gebruikte beveiliging moet bescherming bieden tegen:

- het veranderen van accounting gegevens,
- het verwijderen van accounting gegevens,
- het toevoegen van valse accounting gegevens en
- het ongeoorloofd lezen van accounting gegevens.

Met het genereren van valse accounting berichten zou een bepaalde gebruiker een te hoge rekening kunnen krijgen. Bij het vroegtijdig ontvangen van een accounting−stop bericht zouden, bij verkeerd ontwerp van de accounting server, de opvolgende interim−berichten geweigerd kunnen worden. Hierdoor ontstaat een verkeerd beeld van de sessie. Hetzelfde geldt als legitieme berichten worden veranderd.

Vaak zijn security aspecten in het transport protocol aanwezig. Veiligheid van accounting gegevens is echter niet alleen een onderdeel van het transport protocol, omdat ook via tussenkomst van een accounting proxy moet kunnen worden gewerkt. Er is sprake van end−to−end security (beveiliging van service element tot home accounting server) en hop−by−hop security (beveiliging tussen de verschillende elementen).

Het is nodig om de verschillende niveaus van beveiliging per attribuut te kunnen gebruiken. Het is bijvoorbeeld mogelijk dat bepaalde accounting gegevens alleen voor lokaal gebruik zijn bestemd en niet voor een home server van een ander domein. Ook is het mogelijk dat berichten via untrusted proxies gaan, die de accounting berichten alleen hoeven bezorgen. De bestemming van het accounting bericht moet wel door de proxy vanuit de beschikbare gegevens te bepalen zijn.

Accounting berichten hoeven niet altijd te worden gecodeerd. Als er geen strikte eisen zijn aan de beveiliging vanuit een oogpunt van financieel risico dan kan encryptie achterwege blijven. Dit is bijvoorbeeld het geval als accounting gegevens alleen over een lokaal beschermd netwerk gaan en alleen worden gebruikt voor trend analyse. De mogelijkheid om verschillende vormen van encryptie te gebruiken moet echter wel aanwezig zijn.

4.2.1 Integrity protection (MUST)

Integrity protection wil zeggen dat kan worden geverifieerd of het accounting bericht ongewijzigd is aangekomen. Deze beveiliging moet beter zijn dan de beveiliging tegen bitfouten die in het transport protocol aanwezig is. Ook opzettelijk veranderde accounting gegevens moeten gedetecteerd kunnen worden. Het moet mogelijk zijn om in een accounting bericht de integriteit van het bericht te kunnen garanderen.

Als accounting gegevens via proxies of brokers worden doorgegeven moet de uiteindelijke ontvanger kunnen nagaan welke gegevens van welke instantie afkomstig zijn en of ze daarna niet ongewenst zijn veranderd. Het moet voor proxies echter wel mogelijk zijn om zelf gegevens aan accounting berichten toe te voegen.

Referenties: [ietf−aaa−acct, 5.6], [ietf−nasreq−criteria, 8.4.3.1], [arkko−acctreq, 6.3], [wang−aaa−cel−req, 7], [rfc2477, 4.2.6] en [ietf−mobileip−aaa−reqs, 3.1, 6].

4.2.2 Authenticatie (MUST)

Beide partijen moeten zeker zijn met wie ze te maken hebben. Daarom is het nodig om wederzijdse authenticatie uit te kunnen voeren. Voor de accounting server is het belangrijk te weten dat de accounting gegevens van een legitieme client (service element) afkomstig zijn en voor het service element is het belangrijk te weten of de accounting berichten door een bekende accounting server worden verwerkt.

Referenties: [ietf−aaa−acct, 5.6], [ietf−nasreq−criteria, 8.1.4.1], [arkko−acctreq, 6.3], [ietf−mobileip−aaa−reqs, 3.1] en [wang−aaa−cel−req, 7].

4.2.3 Confidentiality protection (MUST)

Confidentiality protection houdt in dat berichten die zijn bestemd voor een bepaalde ontvanger niet door anderen kunnen worden gelezen. Het moet mogelijk zijn om accounting gegevens te versturen die op deze manier zijn beveiligd. Sommige gegevens in accounting berichten zijn van belang voor proxies om bijvoorbeeld de goede ontvanger van het accounting bericht te bepalen. Andere gegevens moeten dusdanig worden beschermd dat alleen de uiteindelijke ontvanger ze kan lezen.

Referenties: [ietf−aaa−acct, 5.6], [ietf−nasreq−criteria, 8.1.4.4, 8.4.3.1], [arkko−acctreq, 6.3], [wang−aaa−cel−req, 7], [rfc2477, 4.2.6] en [ietf−mobileip−aaa−reqs, 3.1, 6].

4.2.4 Replay protection (MUST)

Het mag niet mogelijk zijn dat een eerder verstuurd accounting bericht of een aanpassing van een eerder verstuurd bericht nogmaals wordt geaccepteerd. Dit beveiligt tegen het opnieuw genereren van accounting gegevens, wat onder andere dubbele billing voorkomt.

Referenties: [ietf−aaa−acct, 5.6] en [ietf−mobileip−aaa−reqs, 3.1].

4.2.5 Non−repudiation (SHOULD)

Non−repudiation betekent dat er niet kan worden ontkend dat een bericht is verzonden of ontvangen. Het service element kan later niet ontkennen dat het accounting bericht is verstuurd en de accounting server kan later niet ontkennen dat hij het bericht heeft ontvangen.

In het accounting protocol moet het mogelijk zijn om non−repudiation van berichten te gebruiken. Dit wordt vooral gebruikt als accounting gegevens tussen verschillende organisaties worden uitgewisseld. Aan accounting gegevens die tussen twee verschillende organisaties worden verstuurd zijn over het algemeen kosten verbonden, wat het onmogelijk maken van ontkennen erg belangrijk maakt.

Referenties: [ietf−nasreq−criteria, 8.4.3.2] en [ietf−mobileip−aaa−reqs, 3.1].

4.2.6 Brokers (MUST)

Het moet mogelijk zijn om brokers te gebruiken bij de communicatie tussen verschillende elementen in het accounting proces. Brokers kunnen voor een beveiligde verbinding zorgen door met beide kanten een beveiligde communicatie op te zetten, maar ook kan een broker worden gebruikt om sleutels voor de encryptie te distribueren.

Referentie: [ietf−mobileip−aaa−reqs, 3.1, 6].

4.3 Accounting event requirements

Accounting event requirements hebben betrekking op het soort berichten die binnen het accounting protocol kunnen worden verstuurd. Niet alle diensten vereisen alle mogelijke soorten accounting berichten. Het accounting protocol moet minimaal de in deze pragraaf genoemde berichten kunnen ondersteunen. Deze berichten komen voort uit de algemene accounting requirements.

Referenties: [ietf−nasreq−criteria, 8.4.1.5], [arkko−acctreq, 6.5] en [arkko−acctrqlis, 4.1].

Er bestaat enige discussie over het feit of accounting gegevens moeten kunnen worden verstuurd voordat de dienst daadwerkelijk wordt geleverd. Dit vooraf genereren van accounting gegevens zou kunnen worden gebruikt om pas dienstverlening toe te staan als dit mogelijk is aan de hand van bijvoorbeeld kredietlimieten. Naar mijn mening is accounting het verslagleggen van gemeten feiten. Dit betekent dat accounting gegevens pas kunnen worden verstuurd als de dienst daadwerkelijk is aangeboden. Het beslissen over verlenen van de dienst is onderdeel van het autorisatiegedeelte. Het autorisatiegedeelte zou op basis van verwacht gebruik een periode kunnen vaststellen waarbinnen de dienst mag worden geleverd. Na afloop van deze periode zou er re−autorisatie moeten plaatsvinden.

4.3.1 Start of a session (start bericht) (MUST)

Aan het begin van het leveren van de dienst wordt vaak een accounting bericht gegenereerd. In zo’n bericht staan in het algemeen zaken over de identiteit van de client, het tijdstip (timestamp) waarop is begonnen met het aanbieden van de dienst, etc. In het start bericht wordt een identificatie aan de sessie gegeven, zodat bij elkaar behorende berichten kunnen worden samengeraapt.

De mogelijkheid tot het versturen van een start bericht moet in het accounting protocol aanwezig zijn. Dit betekent niet dat bij alle diensten waar accounting wordt toegepast start berichten hoeven te worden gestuurd. Voor indivisible events kunnen bijvoorbeeld meteen session records worden gemaakt.

4.3.2 End of a session (stop bericht) (MUST)

Net als bij het begin wordt ook bij de beeindiging van de sessie vaak een accounting bericht gegenereerd. Hierin staan in het algemeen dezelfde gegevens als in het bijbehorende start bericht aangevuld met gegevens over gebruik van de sessie, zoals bijvoorbeeld verstuurde hoeveelheid gegevens, duur van de sessie etc.

Een accounting protocol moet dus zo'n bericht kunnen versturen.

4.3.3 Update of a session (interim bericht) (MUST)

Gedurende de sessie moet het mogelijk zijn om het gebruik tot dan toe door te geven (interim accounting). Dit kan nodig zijn als er een wijziging in de instellingen van de client of de sessie optreedt of bij het verstrijken van een bepaalde hoeveelheid tijd. Interim accounting moet ook het risico van verlies van accounting gegevens bij uitval van het service element minimaliseren.

Het accounting protocol moet het sturen van interim accounting berichten ondersteunen. De accounting server moet het service element kunnen verzoeken om met een bepaald interval interim accounting berichten te sturen.

In interim berichten zouden praktisch dezelfde gegevens komen te staan als in het end bericht, maar dan voor de gegevens die tot nu toe beschikbaar zijn. Wel is het mogelijk om onveranderde gegevens niet op te nemen. Er moet dan aangegeven kunnen worden of het gaat om cumulatieve of incrementele gegevens. Bij gebruik van cumulatieve gegevens wordt al het gebruik na begin van de sessie gemeten, bij gebruik van incrementele gegevens al het gebruik na het vorige interim bericht. Het gebruik van cumulatieve gegevens heeft het voordeel dat bij verlies van een interim bericht geen gegevens verloren gaan.

Soms wachten verschillende interim berichten die betrekking hebben op dezelfde sessie op verzending. Bij gebruik van cumulatieve codering hoeft dan alleen het laatste bericht te worden verstuurd (interim accounting overwrite). Dit kan bijvoorbeeld optreden bij het onbereikbaar zijn van de accounting server. Door alleen het laatste bericht te bufferen kan een aanzienlijke besparing in bufferruimte gerealiseerd worden.

Referenties: [ietf−aaa−acct, 6.1.1, 6.2.2] en [arkko−acctreq, 6.4].

4.3.4 Session record (MUST)

Het moet mogelijk zijn om met het accounting protocol gegevens over de volledige sessie te kunnen versturen. Het zou alle gegevens moeten bevatten die over de gehele sessie bekend zijn. Dit type bericht bevat in principe dezelfde gegevens als het end bericht.

Dit record zou onder andere kunnen worden gebruikt om accounting bij ondeelbare events uit te voeren, maar is ook nodig om gegevens over een sessie te versturen nadat deze al is afgelopen.

4.3.5 Polling (MUST)

Het moet voor de accounting server mogelijk zijn om het service element te vragen gebufferde accounting gegevens te versturen of accounting (interim) gegevens van huidige sessies te genereren.

In geval van uitval van de accounting server moet deze de accounting gegevens die bij de crash verloren zijn gegaan kunnen reconstrueren aan de hand van alle sessies die het service element op dat moment heeft draaien.

Ook is het mogelijk dat de accounting server alle aangesloten service elements met enige regelmaat langsgaat om alle gebufferde accounting gegevens op te halen. Dit heeft als voordeel dat de accounting server de afhandeling van accounting berichten kan verdelen en dat er maar een beperkte hoeveelheid verbindingen nodig zijn. Het opzetten van verbindingen brengt een grote hoeveelheid overhead met zich mee.

Referenties: [ietf−aaa−acct, 6.3.1], [ietf−nasreq−criteria, 8.4.1.6] en [arkko−acctreq, 6.4].

4.3.6 Event−driven polling (MUST)

Bij event−driven polling sturen service elements een bericht naar de accounting server om aan te geven dat er accounting gegevens klaar zijn om te worden verstuurd. De accounting server kan dan, als het hem uit komt, het service element vragen om deze gegevens te versturen. Dit heeft als voordeel dat accounting servers aleen die service elements hoeven te pollen, die accounting gegevens klaar hebben. Verder heeft het, net als bij polling, het voordeel dat er niet voor elk event een nieuwe verbinding opgezet hoeft te worden.

Referenties: [ietf−aaa−acct, 6.3.4] en [arkko−acctreq, 6.4].

4.3.7 Bevestiging van bericht (MUST)

In het algemeen worden accounting berichten bevestigd door de accounting server. Dit is om het service element de zekerheid te geven dat het bericht goed is aangekomen. Dit is ook nodig om aan veel van de security requirements te voldoen.

Referenties: [ietf−nasreq−criteria, 8.4.1.1] en [ietf−mobileip−aaa−reqs, 3.1].

4.3.8 Negotiation of transfer method and format capabilities (MUST)

Het zou mogelijk moeten zijn om, eventueel via een dialoog tussen de accounting server en het service element, afspraken te maken over een te gebruiken accounting model en de parameters die daarbij van toepassing zijn. Hierbij kunnen bijvoorbeeld zaken als real−time accounting, buffering, push of pull en tijdsintervallen van interim berichten worden geregeld. Deze onderhandeling zal in het algemeen plaatsvinden bij het aanmelden van het service element bij de accounting server.

Ook moet het mogelijk zijn om, als het accounting protocol daarin voorziet, afspraken te maken over het te gebruiken transport protocol en record−format en de parameters die daarbij worden gebruikt. Veel van deze zaken zijn natuurlijk off−line in te stellen. Het zou de flexibiliteit van het accounting protocol ten goede komen als dit on−line zou kunnen gebeuren.

Referenties: [arkko−acctreq, 6.4]. en [blount−acct−service, 5].

4.4 Transport requirements

Hier volgen de eisen die aan het transporteren van accounting berichten worden gesteld. Het is mogelijk om voor een accounting protocol meerdere transportprotocollen te definiëren die in verschillende situaties worden gebruikt. Zo ligt bijvoorbeeld een TCP gebaseerd protocol meer voor de hand voor het versturen van grote hoeveelheden gegevens en is UDP bijvoorbeeld geschikt om snel en eenvoudig kleine berichten te versturen.

4.4.1 Betrouwbaar transport (MUST)

Accounting gegevens moeten zelfs met packet−loss, netwerk problemen of het uitvallen van service elements of accounting servers, betrouwbaar worden getransporteerd. Fouten die optreden bij het versturen of berichten die niet aankomen moeten worden gedetecteerd en de verloren gegevens moeten opnieuw worden verstuurd. In het algemeen wordt met bevestigde berichten gewerkt. Het service element moet weten dat de accounting server de gegevens juist heeft ontvangen.

Als een onbetrouwbaar transport protocol (bijvoorbeeld UDP) wordt gebruikt, moet er een retry−mechanisme worden gedefinieerd.

Referenties: [ietf−aaa−acct, 5.6, 6.1], [ietf−nasreq−criteria, 8.1.2.2, 8.4.1.1], [ietf−mobileip−aaa−reqs, 3.1] en [ekstein−nasreq−protocomp, 2.5].

4.4.2 Ondersteuning grote berichten (MUST)

Het moet mogelijk zijn om grote berichten te versturen. Deze eis komt niet alleen uit het batch−requirement, maar ook uit het feit dat accounting records op zich erg groot moeten kunnen zijn.

Dit betekent bijvoorbeeld dat, als UDP als basis voor het transportprotocol wordt gebruikt, er moet worden voorzien in het opdelen van de accounting berichten in verschillende UDP pakketten en dat deze pakketten bij de ontvanger weer in de juiste volgorde moeten worden gereconstrueerd.

4.4.3 Snel van server veranderen (MUST)

Als de primaire accounting server niet meer bereikbaar is moet het mogelijk zijn om snel over te stappen op het versturen van accounting berichten naar een eventuele secundaire accounting server. Deze overstap kan worden gedaan bij het verliezen van de primaire verbinding of het optreden van een timeout bij het versturen van accounting gegevens.

Systemen kunnen worden uitgerust met meerdere accounting servers om uitval van een enkele server of uitval van apparatuur op te vangen of om de belasting te verdelen (load balancing).

Referenties: [ietf−aaa−acct, 6.1.4, 6.1.5] en [ietf−nasreq−criteria, 8.1.2.1, 8.1.2.4].

4.4.4 Buffering van accounting gegevens (MUST)

Als er geen accounting server te bereiken is, moet het protocol eventuele aanwezige buffer−ruimte kunnen gebruiken om accounting gegevens op te slaan, totdat er weer een accounting server te bereiken is. Bij voorkeur moet deze buffer zich op non−volatile storage bevinden. Deze eis komt voort uit de archival accounting requirement.

Het protocol kan regelen dat verschillende update berichten die bij dezelfde sessie horen worden samengenomen om bufferruimte en netwerkbandbreedte (als de verbinding weer tot stand komt) te besparen. Bij het volledig uitvallen van het netwerk of de accounting server is het dienstverlenende gedeelte vaak ook uitgevallen. Bij uitval van de dienstverlening worden natuurlijk niet veel accounting gegevens geproduceerd. Er is dan alleen bufferruimte nodig om van alle tot dantoe lopende sessies gegevens op te slaan.

Referentie: [ietf−aaa−acct, 6.1.5, 6.3.3], [arkko−acctreq, 6.1].

4.4.5 Bidirectionele communicatie (MUST)

Accounting gegevens gaan van service element naar accounting server, maar de accounting server stuurt ook bevestigingen en kan een verzoek doen om gebufferde accounting gegevens op te vragen.

Het transport protocol moet voorzien in bidirectionele communicatie en het moet mogelijk zijn om vanuit beide partijen een verbinding op te zetten.

Referentie: [ekstein−nasreq−protocomp, 2.3].

4.4.6 Flow control (MUST)

Het transport protocol moet zorgen dat berichten het netwerk of de server niet overspoelen. Dit betekent dat bijvoorbeeld bij gebruik van UDP flow control expliciet moet worden geïmplementeerd.

Referenties: [ietf−nasreq−criteria,8.1.2.5] en [ekstein−nasreq−protocomp, 2.5].

4.5 Record format requirements

Het record format beschrijft het formaat dat wordt gebruikt om de accounting berichten te verpakken voor transport en opslag. Accounting gegevens bestaan uit gegroepeerde meetgegevens waarbij een waarde aan een attribuut wordt gekoppeld. Gemeten gegevens worden per sessie gegroepeerd en in een record vastgelegd.

Het record format moet niet afhankelijk zijn van het gebruikte transport protocol. Het record format moet namelijk ook gebruikt kunnen worden om accounting data,voor langere tijd, in een file op te slaan.

In het record format wordt in het algemeen ook vastgelegd om wat voor soort accounting bericht het gaat.

Het kan zijn dat er meerdere record formats zijn gedefinieerd die kunnen worden gebruikt. Het is daarom handig om het record format herkenbaar te maken door in het begin van het record format een identificatie van het record format op te nemen (bijvoorbeeld met de versie van het gebruikte formaat).

4.5.1 Tagged and typed data (MUST)

Accounting berichten bevatten getypeerde gegevens waaraan een naam van een attribuut is verbonden. Als voorbeeld kan aan het attribuut user−id de waarde "arthur@west.nl" worden gekoppeld of transferred−octets met als waarde "125".

Het record format moet gegevens in deze vorm kunnen bevatten. De typering van de data kan voortkomen uit de definitie van de namen van de attributen, maar zou ook in het record format zelf kunnen worden opgeslagen. Welke attributen zich daadwerkelijk in een accounting record bevinden hangt in het algemeen af van de gebruikte dienst.

Referenties: [ietf−nasreq−criteria, 8.1.3] en [blount−acct−service,4.2].

4.5.2 Standaard datatypen (MUST)

In het record formaat moeten een aantal standaardtypen aanwezig zijn. De minimale verzameling van benodigde datatypen bestaat uit

- string,
- integer en
- timestamp.

Met een string kunnen in principe alle mogelijke waarden worden vastgelegd. Het moet mogelijk zijn om tekst als waarde te gebruiken. Strings worden bijvoorbeeld gebruikt om namen van gebruikers vast te leggen.

Met het integer type kan een geheel getal worden vastgelegd, bijvoorbeeld voor het vastleggen van de hoeveelheid getransporteerde bytes. De codering van een getal zou kunnen gebeuren in de vorm van een string, maar bij gebruik van een binair record format is er een efficiëntere codering van integers mogelijk.

Een timestamp wordt gebruikt om een moment vast te leggen. Dit zou in principe met een integer kunnen gebeuren (de hoeveelheid verstreken seconden sinds een bepaalde datum op een bepaalde plaats), maar een afspraak over de te gebruiken codering is aanbevolen. Timestamps moeten in verschillende tijdzones bruikbaar zijn.

Sommige diensten vereisen nog meer vastgelegde datatypen, zoals een internet adres. Deze zijn over het algemeen afhankelijk van de dienstverlening en zouden niet tot de basis van de datatypen moeten behoren. Met strings zijn alle mogelijke datatypen te coderen. De daadwerkelijke betekenis van een attribuut wordt niet in het record format vastgelegd, maar op de plek waar deze worden gebruikt.

Sommige documenten bevelen een apart datatype duration aan die een bepaalde tijdsduur kan aangeven. Deze is echter zonder problemen door een integer weer te geven en behoort dus niet tot de minimaal gewenste datatypen. Het kan mogelijk zijn om meerdere datatypen te definiëren.

Referenties: [blount−acct−service, 4.2.1] en [ietf−nasreq−criteria, 8.1.3.1, 8.4.1.4].

4.5.3 Extensible (MUST)

Het record format moet voorzien in de uitbreiding van de mogelijke attributen, diensten en berichten. Het moet mogelijk zijn om leverancier−afhankelijke attributen in een accounting bericht op te nemen, zonder dat dit het hele accounting bericht verstoort.

Het record format moet ook zijn voorbereid op uitbreidingen die in een eventueel volgende versie worden aangebracht. Het protocol moet zo ontworpen worden dat, bij eventuele uitbreidingen, oude en nieuwe versies door elkaar kunnen worden gebruikt.

Referenties: [ietf−nasreq−criteria, 8.1.3.3], [arkko−acctreq, 6.1], [rfc2477, 4.3] en [ekstein−nasreq−protocomp, 2.3].

4.5.4 Gegroepeerde of gestructureerde attributen (MAY)

Het groeperen of brengen van structuur in attributen is een concept dat bij programmeertalen lang wordt toegepast. Gerelateerde data zouden in een enkele structuur kunnen worden gevat. Ook zouden attributen kunnen worden gegroepeerd om routing van deze gegevens te vereenvoudigen. Bepaalde attributen zouden bestemd kunnen zijn voor verschillende accounting servers waarbij het opsplitsen van gegevens eenvoudiger zou verlopen. Ook zouden accounting attributen kunnen worden gegroepeerd, omdat ze een verschillende vorm van beveiliging kennen. Accounting gegevens kunnen van verschillende bronnen afkomstig zijn, en met verschillende sleutels zijn gecodeerd. Ook zouden bepaalde gegevens voor andere ontvangers verschillend kunnen worden versleuteld.

Een record format zou de mogelijkheid moeten kunnen bieden om accounting attributen te groeperen of op een andere manier structuur aan accounting attributen te geven.

Referentie: [thoughtsmail].

4.5.5 Human readable (MAY)

Er wordt de voorkeur gegeven aan een record format dat human readable is. Dit wil zeggen dat het record format uit tekstregels bestaat. Het gebruik van leesbare records maakt het debuggen eenvoudiger.

4.5.6 Compact record format (SHOULD)

Het formaat moet zo compact mogelijk zijn. Het moet mogelijk zijn met relatief eenvoudige hardware accounting berichten te genereren. Een compact record format reduceert ook de overhead van het accounting protocol.

Referentie: [rfc2477, 4.3].

4.5.7 Uitbreidbare berichten (MUST)

Met het record format moet het mogelijk zijn om alle mogelijke verschillende berichten vast te leggen, die in de bericht requirements zijn genoemd. Dit betekent dat er in het record format moet worden aangegeven of het om een start, stop of ander bericht gaat.

Het is aan te bevelen de verzameling van mogelijke berichten uitbreidbaar te maken om uitbreidingen in de toekomst mogelijk te maken.

4.5.8 Verschillende diensten (MUST)

Het moet mogelijk zijn om het record format te gebruiken om accounting gegevens van verschillende soorten diensten vast te leggen. Het protocol moet niet zijn toegespitst op een enkele soort dienst. Er moet dus een referentie naar een gebruikte dienst worden opgenomen. Het is aan te bevelen om bij het definiëren van diensten versienummers te gebruiken. Dit zorgt voor een standaardisatie van naamgeving van diensten en een manier van omgaan met compatibiliteit.

Referentie: [blount−acct−service, 4.4.2].

4.5.9 Definitie diensten (SHOULD)

Om een algemeen accounting protocol voor verschillende diensten te kunnen gebruiken moeten de diensten voor het accounting protocol bekend zijn. Omdat er nieuwe diensten ontstaan moet het mogelijk zijn om dynamisch deze diensten te kunnen definiëren. Hier moet dan geen lange standaardisatieweg aan vooraf hoeven te gaan.

Het record format zou in het definiëren van diensten kunnen voorzien. Dit betekent dat hetzelfde record format zowel kan worden om accounting gegevens te transporteren als om diensten te definiëren.

Het verdient aanbeveling om diensten niet in het protocol te definiëren, maar eerder een framework te bieden waarvan met elders gedefinieerde diensten gebruik kan worden gemaakt. Zo hoeft de definitie van een accounting protocol niet te veranderen als er een nieuwe dienst wordt toegevoegd. Het kan nodig zijn om van enkele bekende wijdverbreide diensten standaarden van vast te leggen, zodat deze uitwisselbaar zijn.

Referenties: [blount−acct−service, 4.4, 4.4.1, 4.4.2].

4.5.10 Samengestelde diensten (MUST)

Het moet mogelijk zijn om aan te geven dat bepaalde accounting berichten onderdeel zijn van een sessie op een hoger niveau. Hierbij worden accounting berichten van verschillende sessies gegroepeerd rond een andere (virtuele) sessie. Zo behoren individuele verbindingen die worden gebruikt bij teleconferencing tot een enkele teleconferencingsessie.

Referentie: [blount−acct−service 4.4.3].

5. Protocollen

In dit hoofdstuk zal worden ingegaan op de meest voorkomende accounting protocollen. Ook zullen enkele accounting protocollen in ontwikkeling worden behandeld. Veel van de hier behandelde protocollen zijn AAA protocollen, maar bij het behandelen zal de nadruk worden gelegd op het accounting gedeelte. Ook zullen enkele protocollen worden behandeld die (tot op heden) geen accounting functionaliteit kennen, maar wel aan accounting zijn gerelateerd.

Van alle protocollen zal een korte beschrijving worden gegeven van het gebruikte transportprotocol, het record format, de gedefinieerde attributen (voor zover van toepassing) en eventueel security eigenschappen.

5.1 RADIUS

RADIUS (Remote Authentication Dial In User Service) is een protocol dat in [rfc2138] wordt beschreven. Dit protocol wordt gebruikt om authenticatie−, authorisatie− en configuratiegegevens te transporteren tussen een Network Access Server (NAS) en een RADIUS server. RADIUS is ontwikkeld door Livingston Enterprises voor het besturen van hun PortMaster serie van Access Servers en wordt in veel inbelproducten ondersteund. Er is ook veel (gratis) software voor beschikbaar. Later is er aan RADIUS een accounting extensie toegevoegd [rfc2139].

De NAS (het service element) vraagt aan de RADIUS server of het een dienst aan een bepaalde gebruiker mag aanbieden. De RADIUS server heeft een centrale database met gebruikers, wachtwoorden en configuratiegegevens van de gebruikers en beantwoort verzoeken van de NAS.

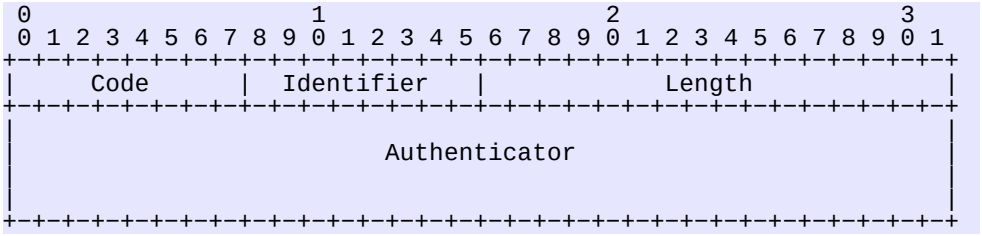
RADIUS gebruikt een binair formaat om zijn gegevens te transporteren en is dus relatief eenvoudig te genereren door simpele hardware. RADIUS gebruikt UDP als transportprotocol. RADIUS pakketten zijn door hun geringe grootte in één UDP pakket te versturen. Omdat er met UDP geen verbinding wordt gemaakt, definieert RADIUS zelf een retry−mechanisme. De invulling van het retry−mechanisme wordt echter aan de implementatie overgelaten.

Een NAS die RADIUS gebruikt is de client van een RADIUS server. De NAS zorgt voor het aanleveren van inloggegevens aan de server en het opvolgen van bevelen van de server. De server zorgt voor het authentificeren van de gebruiker (aan de hand van gegevens van de NAS) en het leveren van configuratiegegevens voor de sessie.

RADIUS transacties worden geauthentificeerd met behulp van een shared secret. User passwords worden versluierd tussen NAS en RADIUS server verstuurd. Overige gegevens worden ongecodeerd verzonden en zijn dus wel gewoon leesbaar.

RADIUS pakketten worden verstuurd via UDP. Dit betekent dat berichten niet te groot kunnen zijn. RADIUS gebruikt zijn eigen timing voor het versturen van retries. Als een eerste RADIUS server niet is te bereiken moet een tweede server kunnen worden bereikt. RADIUS is stateless. Dit alles maakt het gebruik van UDP voor RADIUS meer voor de hand liggend dan TCP. Als bij het versturen niet binnen een bepaalde tijd een bevestiging wordt ontvangen, wordt het bericht een aantal malen herhaald, eventueel bij een andere RADIUS server.

RADIUS berichten bestaan uit een header gevolgd door Attribute−Length−Vaulue 3−tuples. De RADIUS header wordt beschreven in Figuur 5.1.



Figuur 5.1: RADIUS header.

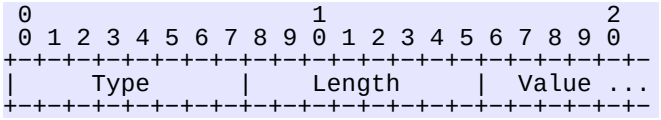
Het Code veld in de RADIUS header geeft het type bericht aan. De mogelijke soorten berichten staan gedefinieerd in Tabel 5.1. Accounting berichten worden verpakt in een Accounting−Request. Deze worden in het algemeen door de RADIUS server beantwoord met een Accounting−Response.

Code	*type of packet*
1	Access−Request
2	Access−Accept
3	Access−Reject
4	Accounting−Request
5	Accounting−Response
11	Access−Challenge
12	Status−Server (experimental)
13	Status−Client (experimental)
255	Reserved

Tabel 5.1: RADIUS Codes. De soorten berichten die in RADIUS gedefinieerd zijn.

Het Identifier veld in de RADIUS header wordt gebruikt om requests en replies bij elkaar te passen. Het Length veld geeft de totale lengte van het RADIUS bericht aan (header en attributen). Het Autenticator veld wordt gebruikt om antwoorden van de RADIUS server te authentificeren en om wachtwoorden te versluieren.

RADIUS attributen bevatten informatie over authenticatie, authorisatie en accounting. De Attributen hebben het volgende formaat:



Figuur 5.2: RADIUS attribute format.

Het Type veld geeft de naam van het attribuut (en daarmee het type van het Value veld). Het Length veld geeft de lengte van dit attribuut (Type, Length en Value velden) aan. Het Value veld geeft de waarde van het attribuut aan. Het type van de Value wordt bepaald aan de hand van de Type en Length velden. In RADIUS zijn de datatypen string, address (32−bit), integer (32−bit) en time (32−bit) gedefinieerd. In Tabel 5.2 staan de attributen die in RADIUS zijn gedefinieerd.

Code	*Attribute name*	*Type*	*In accounting*
2	User−Password	string	0
3	CHAP−Password	CHAP Ident+string	0
4	NAS−IP−Address	address	0−1
5	NAS−Port	integer	0−1
6	Service−Type	integer	0−1
7	Framed−Protocol	integer	0−1
8	Framed−IP−Address	address	0−1
9	Framed−IP−Netmask	address	0−1
10	Framed−Routing	integer	0−1
11	Filter−Id	string	0
12	Framed−MTU	integer	0−1
13	Framed−Compression	integer	0
14	Login−IP−Host	address	0
15	Login−Service	integer	0−1
16	Login−TCP−Port	integer	0−1
18	Reply−Message	string	0
19	Callback−Number	string	0−1
20	Callback−Id	string	0−1

Code	*Attribute name*	*Type*	*In accounting*
22	Framed−Route	string	0
23	Framed−IPX−Network	integer	0−1
24	State	string	0
25	Class	string	0
26	Vendor−Specific	vendor−id+string	0
27	Session−Timeout	integer	0−1
28	Idle−Timeout	integer	0−1
29	Termination−Action	integer	0−1
30	Called−Station−Id	string	0−1
31	Calling−Station−Id	string	0−1
32	NAS−Identifier	string	0−1
33	Proxy−State	string	0
34	Login−LAT−Service	string	0−1
35	Login−LAT−Node	string	0−1
36	Login−LAT−Group	string	0−1
37	Framed−AppleTalk−Link	integer	0−1
38	Framed−AppleTalk−Network	integer	0−1
39	Framed−AppleTalk−Zone	string	0−1
40	Acct−Status−Type	integer	1
41	Acct−Delay−Time	integer	0−1
42	Acct−Input−Octets	integer	0−1
43	Acct−Output−Octets	integer	0−1
44	Acct−Session−Id	string	1
45	Acct−Authentic	integer	0−1
46	Acct−Session−Time	integer	0−1
47	Acct−Input−Packets	integer	0−1
48	Acct−Output−Packets	integer	0−1
49	Acct−Terminate−Cause	integer	0−1
50	Acct−Multi−Session−Id	string	0
51	Acct−Link−Count	integer	0
60	CHAP−Challenge	string	0
61	NAS−Port−Type	integer	0−1
62	Port−Limit	integer	0−1
63	Login−LAT−Port	string	0−1
85	Acct−Interim−Interval	integer	0−1

Tabel 5.2: RADIUS attributen. De kolom In accounting geeft aan hoe vaak het attribuut in een Accounting−Request mag voorkomen:

0 *Het attribuut mag niet worden gebruikt in een Accounting−Request*
0+ *Nul of meer keer mag dit attribuut voorkomen*
0−1 *Het attribuut mag nul of een keer voorkomen*
1 *Het attribuut moet precies een keer voorkomen*

Daarnaast zijn er diverse andere attributen voor bijvoorbeeld tunneling of Producent−specifieke attributen gedefinieerd. Met het Acct−Status−Type attribuut wordt het soort accounting bericht aangegeven. In het document [ietf−radius−acct−interim] wordt interim accounting aan radius toegevoegd. In Tabel 5.3 staan de mogelijke berichten die met RADIUS kunnen worden verstuurd.

Value	*type of message*
1	Start
2	Stop
3	Interim Update
7	Accounting On
8	Accounting Off

Tabel 5.3: RADIUS Acct−Status−Type attribute values.

Als de NAS begint met het leveren van een dienst wordt een Start bericht aan de RADIUS accounting server verstuurd met daarin gegevens over te leveren dienst en de gebruiker van de dienst. Aan het einde van het bieden van de dienst verstuurt de NAS een Stop pakket met statistieken over de gebruikte dienst. Tijdens de dienstverlening kunnen Interim Updates worden verstuurd.

RADIUS is ontworpen als AAA protocol voor het regelen van inbelverbindingen. Hier is het wijd verbreid en in feite de standaard. RADIUS mist echter een aantal eigenschappen om geschikt te zijn als algemeen accounting protocol. Bij RADIUS wordt verondersteld dat alle accounting gegevens real−time worden verstuurd. Batching behoort dus niet tot de mogelijkheden, ook al omdat UDP wordt gebruikt. Ook worden de berichten in RADIUS ongecodeerd verstuurd waardoor ze door iedereen leesbaar zijn. RADIUS is ook te beperkt uitbreidbaar om voor algemene accounting toepassingen te kunnen worden gebruikt.

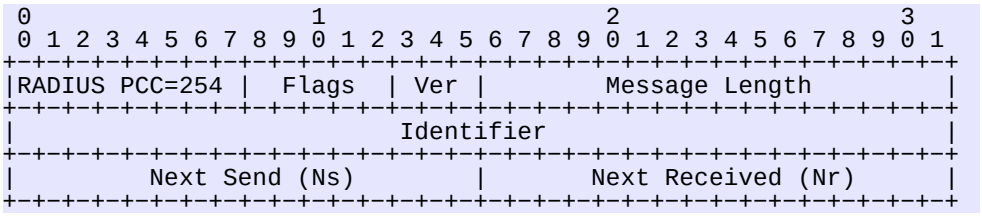
5.2 DIAMETER

Diameter is ontworpen als vervanging van RADIUS. Het DIAMETER framework is gedefinieerd in een Internet draft [calhoun−diameter−framework] die een beschrijving geeft van het ontwerp van DIAMETER. Hierin worden alle problemen en tekortkomingen van RADIUS opgesomd en wordt aangegeven hoe deze met DIAMETER opgelost worden. Het gebruikte record format, het transport protocol en de mogelijke berichten zijn gedefinieerd in [calhoun−diameter]. Bij een toepassing van DIAMETER wordt deze in een apart document als extensie van DIAMETER gedefinieerd. Accounting met DIAMETER is gedefinieerd in [calhoun−diameter−accounting]. Verder zijn er tal van documenten die extensies van DIAMETER beschrijven.

DIAMETER is ontworpen om goed te functioneren in een omgeving met proxies. Het is dan ook goed toepasbaar in een inter−domain omgeving. Het biedt zowel hop−by−hop beveiliging als end−to−end beveiliging.

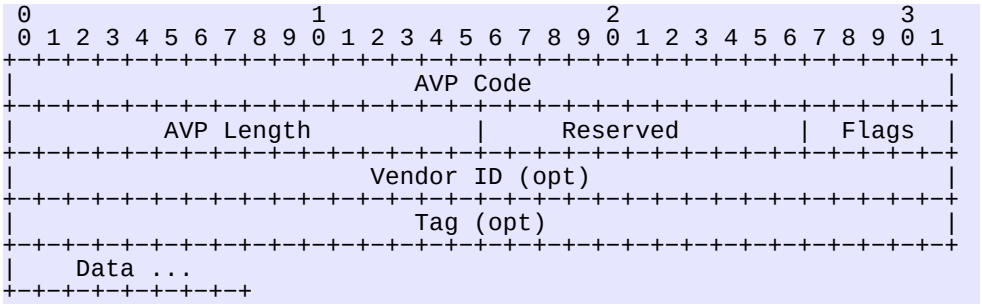
DIAMETER transport verloopt via UDP. In DIAMETER wordt een mechanisme beschreven waarmee flow control geregeld wordt. Met een floating window mechanisme kunnen grote berichten worden verstuurd. Het is dus mogelijk om batching over DIAMETER uit te voeren.

DIAMETER berichten bestaan uit een header beschreven in Figuur 5.3. Het RADIUS Packet Compatibility Code (PCC) is opgenomen om DIAMETER paketten van RADIUS paketten te kunnen onderscheiden. Zo kunnen RADIUS en DIAMETER door elkaar gebruikt worden. Met Flags worden eigenschappen van het transport aangegeven, Message Length geeft de lengte van het bericht aan en Identifier is bedoeld om requests by replies vinden. De Next Send en Next Received worden gebruik als er met een window mechanisme gebruikt wordt.



Figuur 5.3: DIAMETER header.

Na deze header volgen AVP’s zoals beschreven in Figuur 5.4. Deze bevatten attributen die afhankelijk zijn van de toepassing. De AVP Code geeft het attribuut aan. Dit kan een leverancier specifiek attribuut zijn, naar gelang dat in de Flags is aangegeven. De eerste 256 attributen zijn gereserveerd voor RADIUS compatibiliteit en moeten dus ook geinterpreteerd worden zoals dat bij RADIUS gebruikelijk is.



Figuur 5.4: DIAMETER AVP format.

De AVP Length geeft de lengte van de AVP aan en met de Flags worden zaken aangegeven over het attribuut, zoals of het om een leverancier−specifiek attribuut gaat. Met het Vendor ID wordt een eventuele leverancier aangeduid, die het attribuut gedefinieerd heeft en het Tag veld kan gebruikt worden om meerdere attributen die bij elkaar horen te nummeren. Hierna volgt de data. In xxx staan de mogelijke datatypen opgesomd.

Type	***Omschrijving***
Data	binaire data in een willekeurig formaat
String	NULL−terminated variable length UTF−8 string
Address	32 bit (IPv4) of 128 bit (IPv6) adres, afhankelijk van het AVP Length veld
Integer32	32 bit waarde
Integer64	64 bit waarde
Time	32 bit unsigned geeft het aantal verstreken seconden aan sinds 00:00:00 GMT, 1 Januari 1900
Complex	complexe datattypen

Tabel 5.4: DIAMETER datatypen.

Accounting bij DIAMETER kan worden uitgevoerd door attributen van RADIUS te gebruiken. Het is echter ook mogelijk om, met een ADIF−Record attribuut accounting gegevens in ADIF formaat te versturen. Met het Accounting−Record−Type attribuut wordt dan het soort accounting bericht aangegeven. De soorten berichten staan gegeven in Tabel 5.5.

Value	***Name***	
1	EVENT_RECORD	ondeelbare event
2	START_RECORD	start van dienstverlening
3	INTERIM_RECORD	tijdens de dienstverlening
4	STOP_RECORD	einde van de dienstverlening

Tabel 5.5: DIAMETER verschillende berichten.

De DIAMETER server kan aan de client (het service element) aangeven hoe vaak interim berichten moeten worden verstuurd en hoeveel accounting gegevens gebufferd moeten worden. Tijdens de autorisatie van de dienstverlening kan met het Accounting−Interim−Interval attribuut de frequentie van de interimberichten, met Delivery−Max−Batch de maximum hoeveelheid van de gebufferde accounting gegevens en met Deliverey−Max−Delay de maximale wachttijd voor accounting berichten aangegeven worden.

5.3 COPS

Het COPS (Common Open Policy Service) protocol is een eenvoudig vraag en antwoord protocol in een client/server omgeving, dat gebruikt wordt om informatie over policies uit te wisselen tussen een policy server en haar clients. COPS is ontworpen om autorisatie te regelen bij RSVP resource requests in netwerken die Intserv ondersteunen. Het protocol is zo ontworpen om toegang tot algemene resources (diensten) te kunnen regelen, maar heeft op het moment geen accounting deel gedefinieerd. COPS is beschreven in [ietf−rap−cops].

Een Policy Decision Point (PDP) fungeert als policy server. Een Policy Enforcement Point (PEP) bevind zich in het service element en doet autorisatie requests bij een PDP. De PEP voert dan de beslissingen van de PDP uit. Hierbij kan de PEP een statusbericht aan de PDP sturen, met daarin informatie over de uitgevoerde besluiten. Deze statusgegevens zouden voor accounting toepassingen gebruikt kunnen worden.

Het protocol gebruik TCP als transport protocol en voorziet in beveiliging op bericht niveau met authenticatie, replay protection en integrity protection.

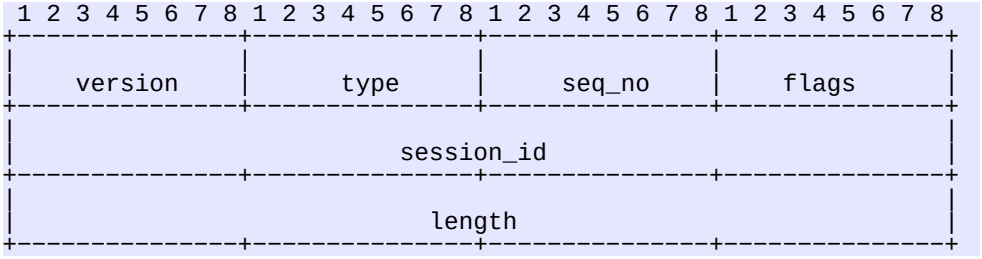
5.4 TACACS+

TACACS+ is de meest recente opvolger van TACACS. TACACS staat voor Terminal Access Controller Access Control System. TACACS+ versie 1.78 is gedefinieerd in [grant−tacacs].

TACAS is een eenvoudig UDP gebaseerd protocol om access control te bieden. TACACS is oorspronkelijk ontwikkeld door BBN voor het MILNET. Cisco heeft een aantal verbeteringen aan TACACS aangebracht. De implementatie van Cisco wordt ook wel XTACACS genoemd.

In TACACS+ worden de authenticatie autorisatie en accounting functies gescheiden. Tevens wordt al het verkeer tussen de NAS (service element) en de AAA−server gecodeerd verzonden. Het ondersteunt ook grotere berichten en meerdere vormen van authenticatie. Het is uitbreidbaar om eigen attributen en uitbreiding in de toekomst te ondersteunen. Het gebruikt TCP in plaats van UDP als onderliggend transportprotocol, om de betrouwbaarheid te verhogen TACACS+ ondersteund zowel de klassieke user/password authenticatie als one−time passwords en challenge−respons authenticatie en kan bij de autorisatie specifieke configuratiegegevens doorgeven. Een voorbeeld van beperkte autorisatie is het instellen van een tijdbeperking op de verbinding.

TACACS+ heeft een binair record−format. Alle TACACS+ berichten worden voorafgegaan door de packet header die beschreven is in Figuur 5.5. De header zelf wordt niet gecodeerd verzonden en beschrijft de rest van het bericht. Met het version veld wordt de gebruikte versie van TACACS+ aangegeven. Met het type veld wordt aangegeven of het om een authenticatie, autorisatie of accounting bericht gaat. TACACS+ sessies zijn enkele gevallen van het uitvoeren van authenticatie, een autorisatie request of het uitwisselen van een accounting bericht. Met het seq_no veld worden berichten die tot dezelfde TACACS+ sessie behoren oplopend genummerd. Met de flags wordt aangegeven of encryptie en multiplexing wordt toegepast. Het session_id geeft een uniek nummer aan de TACACS+ sessie. Het length veld geeft de lengte van het TACACS+ bericht aan.

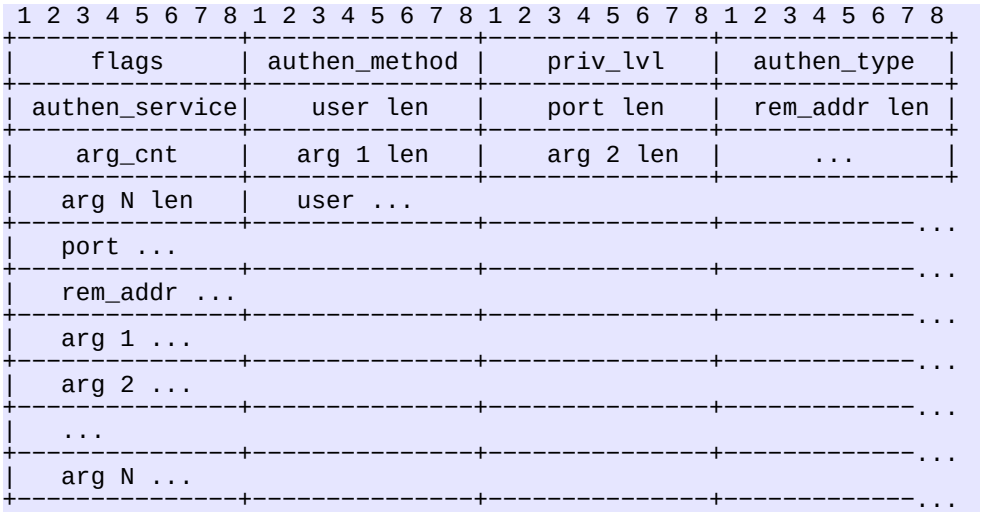


Figuur 5.5: TACACS+ packet header.

De inhoud van een TACACS+ bericht wordt gecodeerd door MD5−hashes over het gegevens uit de header en een sleutel, te gebruiken als een pseudo−random string. De gemeenschappelijke secret key is alleen bij zender en ontvanger bekend.

In TACACS+ bestaan accounting berichten uit accounting requests en accounting replies. De body van het accounting request formaat staat beschreven in Figuur 5.6, die van de reply in Figuur 5.7. Met flags wordt het soort bericht aangegeven (start, stop of watchdog), het authen_method veld geeft de gebruikte authenticatiemethode aan, het priv_lvl geeft de huidige gebruikersprivileges aan, Het authen_service geeft de dienstverlening aan (zie Tabel 5.6). De len velden geven de lengte van de volgende attributen aan. Het user veld geeft de gebruikersnaam aan en het

port veld de naam van de poort waarop de verbinding plaatsvindt. Daarna volgen de attributen. Deze zijn afkomstig uit het autorisatie gedeelte, aangevuld met enkele specifieke attributen. De mogelijke attributen staan vermeld in Tabel 5.7. Een TACACS+ accounting reply bevat een status (success, error of follow), een een bericht van de server voor de gebruiker en een bericht voor een eventuele systeembeheerder.



Figuur 5.6: TACACS+ accounting request packet body.

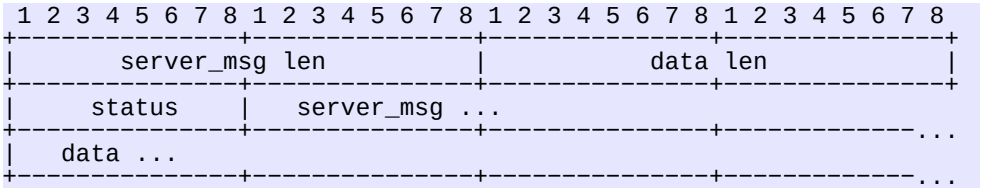
Value	***Name***	
0x00	TAC_PLUS_AUTHEN_SVC_NONE	onbekende dienst
0x01	TAC_PLUS_AUTHEN_SVC_LOGIN	
0x02	TAC_PLUS_AUTHEN_SVC_ENABLE	
0x03	TAC_PLUS_AUTHEN_SVC_PPP	
0x04	TAC_PLUS_AUTHEN_SVC_ARAP	
0x05	TAC_PLUS_AUTHEN_SVC_PT	
0x06	TAC_PLUS_AUTHEN_SVC_RCMD	
0x07	TAC_PLUS_AUTHEN_SVC_X25	
0x08	TAC_PLUS_AUTHEN_SVC_NASI	
0x09	TAC_PLUS_AUTHEN_SVC_FWPROXY	

Tabel 5.6: TACACS+ verschillende diensten.

Attribute name	***Omschrijving***
service	soort dienst
protocol	het gebruikte protocol in de dienstverlening
cmd	het uitgevoerde commando
cmd−arg	argumenten bij het commando
acl	connection access list
inacl	interface input access list
outacl	interface output access list
zonelist	appletalk attribuut
addr	netwerk adres
addr−pool	groep adressen
routing	wordt routing informatie doorgegeven

Attribute name	*Omschrijving*
route	geeft een gebruikte route aan
timeout	gebruikte timeout
idletime	gebruikte idletime
autocmd	commando wat in een eventuele shell gedraaid wordt
noescape	gebruiker mag geen escapes gebruiken
nohangup	verbreek verbinding niet na automatisch commando
priv_lvl	het privilege niveau
remote_user	
remote_host	
callback−dialstring	het gebruikte nummer om terug te bellen
callback−line	
callback−rotary	
nocallback−verify	
task_id	de id die bij de sessie hoort
start_time	begin van de sessie
stop_time	de tijd dat de sessie eindigde
elapsed_time	de verlopen tijd in seconden
timezone	de timezone voor alle timestamps in dit bericht
event	
reason	rede voor optreden van de event
bytes	het aantal bytes dat bij de sessie getransporteerd is
bytes_in	
bytes_out	
paks	het aantal packets dat bij de sessie getransporteerd is
paks_in	
paks_out	
status	geeft status aan
err_msg	foutmelding

Tabel 5.7: TACACS+ attributen.



Figuur 5.7: TACACS+ accounting reply packet body.

5.5 SNMP

SNMP (Simple Network Management Protocol) is een wijdverbreid communicatie protocol dat gebruikt wordt voor het beheer van allerlei, voornamelijk TCP/IP gebaseerde, applicaties [snmpintro]. SNMP is gedefinieerd in een groot aantal RFC's. SNMP wordt uitgebreid met MIB's (Management Information Base), waar definities van objecten en hun betekenis in staan. Er zijn zeer veel van deze MIB's voor handen voor allerlei verschillende apparatuur.

SNMP kan over meerdere protocollen getransporteerd worden, maar wordt in het algemeen over UDP getransporteerd. Het is niet ontworpen als accounting protocol, maar kan daarvoor, binnen bepaalde grenzen, wel voor gebruikt worden. SNMP kent vier soorten berichten:

- een get request,
- een get next request,

- een set request en
- een trap message.

Met een get request kan een waarde uit een SNMP object opgevraagd worden. Met behulp van een get next request kunnen alle meetbare waarden in een SNMP object nagelopen worden. Het set request wordt gebruik om instellingen te veranderen. Een trap message kan door een object worden gegenereerd als een bepaalde toestand optreed. Accounting informatie zou via polling via get requests of een trap−mechanisme verstuurd kunnen worden.

SNMP gebruikt ASN.1 als formaat om gegevens te representeren. ASN.1 is een gestandariseerd binair formaat om berichten mee vast te leggen.

5.6 MSIX

Metered Service Information eXchange (MSIX) (spreek uit als M6 in het Engels) is een protocol dat is ontwikkeld om gegevens over het gebruik van communicatie−diensten (bijvoorbeeld inbelfaciliteiten van een internet provider) te beschrijven en te transporteren. MSIX is echter niet beperkt tot een bepaalde dienst of een beperkte verzameling van diensten.

MSIX is ontwikkeld door NetCentric en Compaq met hulp van zo'n twintig andere bedrijven [msixwww]. De eerste revisie is gepresenteerd op 1 juni 1997. In 1999 is het werk aan MSIX opgepakt door MatraTech en onder de hoede van de Internet Engineering Task Force (IETF) gebracht. MSIX is momenteel beschreven in een Internet draft [blount−acct−msix].

MSIX is gebaseerd op XML [xmlw3c] (Extensible Markup Language) en is dus ook human−readable. XML is wel een formaat wat veel tekst gebruikt om relatief eenvoudige dingen te beschrijven, maar door gebruik van compressie valt de grootte flink te reduceren. MSIX is een record−format dat voorziet in het definiëren van services en het versturen van gebruik van deze services. Het is heeft geen authenticatie en autorisatie gedeelten en is dus geen algemeen AAA protocol.

Een belangrijke eigenschap van MSIX is dat er 'compound' services en sessions gedefinieerd kunnen worden. Hierin kan een parent service gedefinieerd worden waaronder verschillende child services hangen. Een voorbeeld hiervan is teleconferencing waarbij verschillende verbindingen gebruikt worden. Deze kunnen dan onder een noemer bijgehouden worden.

Bij MSIX is geen transportprotocol gedefinieerd, maar wordt het gebruik van een HTTP/SSL/TCP/IP stack gesuggereerd. De basis van het MSIX bericht wordt in Figuur 5.8 beschreven. De timestamp geeft het moment aan waarop het bericht gegenereerd is, de uid wordt gebruikt om een unieke identificatie aan het bericht te geven.

```
<?xml version=1.0?>
<msix version="1.2" timestamp="2000−01−04T13:38:25Z" uid="[uid]">

    ...

</msix>
```

Figuur 5.8: MSIX bericht.

De MSIX berichten die zijn gedefinieerd, staan vermeld in Tabel 5.8. Alle berichten worden door het service element geïnitieerd. De MSIX server beantwoordt deze berichten. Het status bericht kan aan een bericht (in het algemeen een response van de server) worden toegevoegd om aan te geven of de request correct is ontvangen en afgehandeld. Met deze berichten worden diensten gedefinieerd en accounting berichten beschreven.

Message	
defineservice	definitie van een dienst
beginsession	geeft aan dat dienst begonnen is
updatesession	geeft update over dienst
commitsession	geeft het einde van de sessie aan
abortsession	maak sessie ongedaad (bewaar geen gegevens)
getversions	vraag ondersteunde MSIX versies op
relateservices	breng een parent−child relatie tussen gedefinieerde dientsen aan.
status	geef status (bijvoorbeeld foutmelding) van request aan
defineservicers	antwoord op defineservice
beginsessionrs	antwoord op beginsession
updatesessionrs	antwoord op updatesession
commitsessionrs	antwoord op commitsession
abortsessionrs	antwoord op abortsession
relateservicesrs	antwoord op relateservices
getversionrs	antwoord op getversions

Tabel 5.8: MSIX berichten.

Een voorbeeld van een definitie van een dienst is beschreven in Figuur 5.9. Hierin wordt een telefoondienst beschreven. Dit bericht wordt verstuurd in een MSIX bericht zoals in Figuur 5.8 beschreven. bij het versturen van accounting gegevens wordt een referentie opgenomen naar de gedefinieerde dienst (server.net/FoneCall in ons voorbeeld). In Figuur 5.10 wordt een beginsession bericht gestuurt voor de gedefinieerde sessie. Dit is een start bericht. De MSIX server beantwoord deze met een bericht als beschreven in Figuur 5.11.

```
<defineservice>
    <dn>server.net/Fonecall</dn>
    <version>7.3</version>
    <description>Internet to PSTN telephone call</description>
    <ptype>
        <dn>AccountId</dn>
        <type>STRING</type>
    </ptype>
    <ptype>
        <dn>DialedNumber</dn>
        <type>STRING</type>
    </ptype>
    <ptype>
        <dn>Duration<dn>
        <type>INT32</type>
    </ptype>
    <ptype>
        <dn>StartTime</dn>
        <type>TIMESTAMP</type>
    </ptype>
</defineservice>
```

Figuur 5.9: MSIX service definitie.

```
<beginsession>
    <uid>[sessionid]</uid>
    <dn>server.net/FoneCall</dn>
    <property>
        <dn>AccountId</dn>
        <value>324955</value>
    </property>
    <property>
        <dn>DialedNumber</dn>
        <value>+16177205200</value>
    </property>
    <property>
        <dn>Duration</dn>
        <value>280</value>
    </property>
    <property>
        <dn>StartTime</dn>
        <value>1997-06-06T09:35:22Z</value>
    </property>
</beginsession>
```

Figuur 5.10: eenvoudige MSIX accounting request.

```
<beginsessionrs>
    <status>
        <code>msix.org/200</code>
    </status>
    <uid>[sessionid]</uid>
</beginsessionrs>
```

Figuur 5.11: eenvoudige MSIX accounting response.

5.7 ADIF

ADIF staat voor Accounting Data Interchange Format en is momenteel gedefinieerd in een Internet draft [ietf−roamops−actng]. ADIF is gebaseerd op MIME [rfc1521] en is human−readable. Het is ontworpen om accounting gegevens van een bestaand protocol op een compacte manier in tekst weer te geven. ADIF is dus geen volledig accounting protocol, maar een record format om accounting data uit bestaande protocollen weer te geven.

Een ADIF bericht bestaat uit een header met algemene informatie over de records in het bericht, gevolgd door een aantal records die gescheiden worden door een scheidingsteken. In de header staat het versie nummer, de naam of omschrijving van het service element en een timestamp die het begin van het verzamelen van gegevens aangeeft. Optioneel wordt een protocol aangegeven waaruit de attributen worden gebruikt.

Elk record bestaat uit een of meerdere regels. Zoals in MIME gebruikelijk is kunnen grote strings op de volgende regel vervolgd worden door te beginnen met een spatie of tab. Regels die met het ’#’ symbool beginnen worden als commentaar opgevat. ADIF ondersteunt het gebruik van attributen uit elk ander protocol. Dit gebeurt door attributen uit andere protocollen te gebruiken. Attributen worden aangegeven door de naam van het protocol waaruit ze afkomstig zijn gevolgd door het nummer van het attribuut. Zo geeft `radius//46` het `Acct−Session−Time` attribuut (attribuut 46) uit het RADIUS protocol aan. Als in de header een default protocol is aangegeven kan volstaan worden met het nummer van het attribuut. ADIF voorziet ook in attributen met sub−attributen en het gebruiken van aliassen voor delen van lange object identifiers.

```
version: 1
device: server3
descripton: Accounting Server 3
date: 02 Mar 1999 12:19:01 −0500
defaultProtocol: radius

rdate: 02 Mar 1999 12:20:17 −0500
#NAS−IP−Address
4: 204.45.34.12
#NAS−Port
5: 12
#NAS−Port−Type
61: 2
#User−Name
1: fred@bigco.com
#Acct−Status−Type
40: 2
#Acct−Delay−Time
41: 14
#Acct−Input−Octets
42: 234732
#Acct−Output−Octets
43: 15439
#Acct−Session−Id
44: 185
#Acct−Authentic
45: 1
#Acct−Session−Time
46: 1238
#Acct−Input−Packets
47: 153
#Acct−Output−Packets
48: 148
#Acct−Terminate−Cause
49: 11
#Acct−Multi−Session−Id
50: 73
#Acct−Link−Count
51: 2
```

Figuur 5.12: voorbeeld ADIF file.

5.8 TIPHON

TIPHON (Telecommunications and Internet Protocol Harmonization Over Networks) is een initiatief van de ETSI (European Telecommunications Standards Institute) [etsiwww] om telefoonverkeer over IP en over geschakelde netwerken te combineren. Zo is telefonie mogelijk tussen gebruikers die telefoneren via IP netwerken en gebruikers die aan een geschakeld netwerk zijn aangesloten, zoals PSTN, ISDN of GSM.

Om gegevens over inter−domain pricing, autorisatie en gebruik uit te wisselen is in [ts101321] een op XML gebaseerd protocol beschreven. Gegevens worden via een HTTP verbinding uitgewisseld met gebruikmaking van SSL of TLS beveiliging. TIPHON is echter niet toepasbaar als algemeen accounting protocol, omdat het specifiek is toegespitst op telefoongegevens.

In Figuren 5.13 en 5.14 is een voorbeeld van een Usage Exchange gegeven. De client geeft aan det een telefoonsessie 10 minuten geduurd heeft. De server genereerd hierop een bevestiging.

```
<?xml version=1.0?>
<Message messagId="234565432" random="87654321">
   <UsageIndication componentID="13579990">
      <Timestamp>
         1998−04−24T22:03:00Z
      </Timestamp>
      ...
      <SourceInfo type="e164">
         81458811202
      </SourceInfo>
      <DestinationInfo type="e164">
         4766841360
      </DestinationInfo>
      ...
      <UsageDetail>
         <Amount>
            10
         </Amount>
         <Increment>
            60
         </Increment>
         <Unit>
            s
         </Unit>
      </UsageDetail>
   </UsageIndication>
</Message>
```

Figuur 5.13: vereenvoudigd TIPHON UsageIndication bericht.

```
<?xml version=1.0?>
<Message messagId="234565432" random="87654321">
   <UsageConfirmation componentID="13579990">
      <Timestamp>
         1998−04−24T22:44:00Z
      </Timestamp>
      <Status>
         <Code>
            201
         </Code>
         <Description>
            new usage information created
         </Description>
      </Status>
   </UsageConfirmation>
</Message>
```

Figuur 5.14: vereenvoudigd TIPHON UsageConfimation bericht.

5.9 OFX

OFX staat voor Open Financial eXchange en is in 1997 ontwikkeld door Intuit, Microsoft en Checkfree. OFX is een protocol om financiële gegevens te transporteren en is bedoeld om on−line bankieren en e−commerce dienstverlening over het internet mogelijk te maken. Met OFX kunnen financiële gegevens tussen financiële instellingen, ondernemingen en klanten worden uitgewisseld. De OFX specificatie is beschikbaar op de internet site [ofxwww].

Met OFX wordt direct een prijskaartje aan eventueel gebruik gegeven en is dus niet algemeen geschikt als accounting protocol. OFX kan wel gebruikt worden voor on−line billing als accounting wordt toegepast. OFX ondersteunt een grote diversiteit aan financiele taken, waaronder electronic banking voor kleine en middelgrote klanten, bill payment voor kleine klanten en bedrijven en het sturen van rekeningen.

OFX gebruikt XML om gegevens vast te leggen en is daarmee In figuur Figuur 5.15 is een voorbeeld gegeven waarbij de gebruiker Dan vraagt welke rekeningen sinds 1997−03−01 zijn binnengekomen. In Figuur 5.16 is een sterk vereenvoudigd antwoord op deze vraag weergegeven.

```
<OFX>
  <SIGNONMSGSRQV2>
    <SONRQ>
      <DTCLIENT>19970409090000    <!--Current date 4/9/1997-->
      <USERID>123-45-6789         <!--Dan's user id-->
      <USERPASS>DansPassword
      <LANGUAGE>ENG
      <APPID>EndUserApp
      <APPVER>0700
    </SONRQ>
  </SIGNONMSGSRQV2>
  <PRESDLVMSGSRQV1>
    <PRESLISTTRNRQ>
      <TRNUID>12345
      <PRESLISTRQ>
        <BILLPUB>ABillPublisher
        <DTSTART>19970301000000   <!--Get bills since 3/1/97-->
        <NOTIFYWILLING>Y
        <INCLUDEDETAIL>Y
      </PRESLISTRQ>
    </PRESLISTTRNRQ>
  </PRESDLVMSGSRQV1>
</OFX>
```

Figuur 5.15: OFX bill request..

```
<OFX>
  <PRESDLVMSGSRSV1>
    <PRESLISTTRNRS>
      <PRESLISTRS>
        <BILLPUB>ABillPublisher
        <USERID>123-45-6789
        <DTSTART>19970301000000
        <DTEND>19970409090000
        <PRESLIST>
          <PRESBILLINFO>
            <AMTDUE>124.24      <!--to pay $124.24-->
            <DTPMTDUE>19970501  <!--by 5/1/97 -->
            <BILLDETAILTABLE>
              <TABLENAME>usage
              <BILLDETAILTABLETYPE>x_Power_usage
              <BILLDETAILROW>
                <C>elec         <!--Consumable-->
                <C>19970228     <!--Date meter reading start->
                <C>65543        <!--Meter reading at start-->
                <C>19970328     <!--Date meter reading end-->
                <C>65643        <!--Meter reading at end-->
                <C>100          <!--Difference in readings-->
                <C>KWH          <!--Units-->
                <C>.8934        <!--Rate (price per unit)-->
                <C>89.34        <!--Charge -->
              </BILLDETAILROW>
              <BILLDETAILROW>
                ... another detail row of the bill
              </BILLDETAILROW>
            </BILLDETAILTABLE>
          </PRESBILLINFO>
          <PRESBILLINFO>
            ... another bill
          </PRESBILLINFO>
        </PRESLIST>
      </PRESLISTRS>
    </PRESLISTTRNRS>
  </PRESDLVMSGSRSV1>
</OFX>
```

Figuur 5.16: vereenvoudigde OFX bill respons..

6. Evaluatie protocollen

In dit hoofdstuk worden de behandelde protocollen tegen de requirements gehouden worden om te kijken welke protocollen voldoen. Hier zullen alleen de protocollen behandeld worden die al een accounting deel gedefinieerd hebben.

6.1 RADIUS

RADIUS heeft grote tekortkomingen in de security requirements. RADIUS is ontworpen voor transport tussen service element en RADIUS server en kent alleen beveiliging per verbinding. Er is geen mogelijkheid om, als er sprake is van inter−domain accounting, end−to−end security toe te passen. RADIUS berichten kunnen ook niet versleuteld worden, zodat ze alleen voor de ontvanger leesbaar zijn.

In RADIUS is het transportprotocol maar gedeeltelijk gedefinieerd, wat er voor zorgt dat er verschillende varianten in omloop zijn. Verder is er geen sprake van flow control wat er voor zorgt dat het netwerk verstopt kan raken bij erg veel berichten. RADIUS wordt ook gebruikt bij inter−domain accounting, maar hier is het niet op ontworpen en het kent dan ook nogal wat nadelen op timing gebied en op security gebied.

De schaalbaarheid van RADIUS is beperkt, vooral omdat de eigenschappen van het transport protocol niet zijn vastgelegd. Verder moet per accounting event een bericht verstuurd worden wat er voor zorgt dat er sporake is van overhead per bericht en niet per verzameling berichten.

RADIUS is specifiek ontworpen voor het ondersteunen van inbellen en accounting van andere diensten kan niet eenvoudig gerealiseerd worden. RADIUS is dus niet toepasbaar als algemeen accounting protocol.

Wel moet bij het ontwerpen van accounting protocollen rekening gehouden worden met RADIUS. Het is een protocol wat op veel plaatsen in gebruik is en een globale vervanging door een protocol wat niet met RADIUS samen kan werken zal op veel problemen stuiten.

6.2 DIAMETER

Omdat DIAMETER is ontworpen op basis van de problemen met RADIUS, zijn de security problemen die in RADIUS aanwezig zijn verholpen. Security kan end−to end geschieden of er kan voor een hop−by−hop security aanpak gekozen worden. Ook zijn de tekortkomingen van het transportgedeelte opgelost door het retry−mechanisme uit te breiden met een sliding window mechanisme.

Het blijft een protocol met hetzelfde uitgangspunt als RADIUS: een AAA protocol voor inbellen. Het is dus ook toegespitst op inbellen, hoewel het wel meer faciliteiten voor extensies en leverancier−specifieke attributen biedt. De mogelijkheid om ADIF records in DIAMETER in te bedden is een methode wat accounting van meerdere soorten diensten mogelijk maakt.

DIAMETER, in combinatie met ADIF, een goede kandadaat om RADIUS te vervangen. DIAMETER ondersteund echter geen event−driven polling, accounting gegevens en berichten zijn deels in een binair formaat, deels in ADIF formaat opgeslagen voor de definitie van diensten moet er een aanpassing aan de standaard geschreven worden.

DIAMETER is nog vol in ontwikkeling en wordt nog niet toegepast in systemen. Vooral het accounting gedeelte is nog niet zo oud. Dit alles zorgt er voor dat er nog geen eenduidige implementatie gemaakt kan worden.

6.3 TACACS+

TACACS+ is te vergelijken met RADIUS en wordt dan ook vaak als alternatief voor RADIUS gebruikt. Verschillen zijn het gebruik van TCP in plaats van UDP en het bieden van confidentiality. Ook TACACS+ is geen algemeen accounting protocol en in het ontwerp toegespitst op inbellen met alleen hop−by−hop security.

Verder zijn veel van de beperkingen van RADIUS ook van toepassing op TACACS+. In TACACS+ wordt security hop−by−hop toegepast, zodat het niet toepasbaar is bij untrusted proxies [ietf−aaa−acct, 7.1.2].

6.4 SNMP

SNMP is een niet erg simpel netwerk management protocol en geen accounting protocol. Het kan in enkele gevallen wel als accounting protocol gebruikt worden. Het mist toch enkele eigenschappen zoals end−to−end security. Ook het ontbreken van een goede mogelijkheid om event−driven accoutnign uit te voeren en het onbevestigde karakter van

SNMP berichten zorgt voor beperkte toepasbaarheid. Verder is SNMP een protocol waarmee een grote hoeveelheid aan management specifieke taken mee uitgevoerd kan worden. Een protocol wat zo algemeen is, is moeilijk in te passen op een plek waarbij zulke specifieke requirements spelen als bij accounting.

6.5 MSIX

MSIX is gebaseerd op XML en gebruikt dus veel karakters om een accounting bericht te coderen. MSIX is echter, in tegenstelling tot andere accounting protocollen, wel ontworpen om accounting gegevens over verschillende soorten diensten te transporteren. Met MSIX kunnen diensten gedefinieerd worden en relaties tussen verschillende diensten aangegeven worden.

Bij MSIX wordt HTTP als transport protocol gesuggereerd, wat voor een aanzienlijke overhead zorgt. HTTP is allesbehalve eenvoudig en biedt veel meer functionaliteit dan voor een accounting protocol nodig is. Het is niet waarschijnlijk dat, met gebruikmaking van XML en HTTP, accounting berichten binnen een erg kort tijdsbestek geproduceerd, verstuurd en verwerkt kunnen worden. Een ander nadeel van HTTP is dat de accounting server geen verbindingen kan initieren naar de client voor het doen van polling.

Tijdens het transport worden gegevens via de SSL laag gecodeerd. Dit betekent dat de encryptie hop−by−hop plaatsvind. Er wordt geen ondersteuning geboden voor end−to−end codering.

6.6 ADIF

ADIF is geen accounting protocol, maar een manier om accounting gegevens op te slaan. Daarom kan het ook niet echt met de andere protocollen vergeleken worden. ADIF is gebaseerd op mime, wat human−readable is, terwijl het behoorlijk compact blijft. Met ADIF worden alleen volledige session records opgeslagen en er is dus ook geen sprake van een accounting berichten. ADIF ken geen datatypen, alles wordt als tekst opgeslagen.

ADIF is echter wel te gebruiken als algemeen record−format, omdat het de attributen van andere protocollen gebruikt en op die manier uitstekend uitbreidbaar is. ADIF is daarom een aardige basis om te gebruiken bij het ontwerp van een algemeen accounting protocol.

6.7 Overzicht

In Tabel 6.1 worden de bovengenoemde accounting protocollen tegen de requirements afgezet. Hieruit blijkt dat geen enkel protocol momenteel geschikt is om als algemeen accounting protocol gebruik te worden. Wel is het zo dat een aantal protocollen voor bepaalde toepassingen voldoen.

requirement	***RADIUS***	***DIAMETER***	***TACACS+***	***SNMP***	***MSIX***	***ADIF***
1. Algemene requirements						
1.1 Real−time accounting (MUST)	+	+	+	+	−	n
1.2 Archival accounting (MUST)	−	+	−	−	+	n
1.3 Batch accounting (MUST)	−	+	−	+	+	n
1.4 Minimale overhead (SHOULD)[1]	+	+	+	&	−	&
1.5 Schaalbaar (MUST)	−	+	−	+	+	n
1.6 Ondersteuning van eindige sessies (MUST)	+	+	+	+	+	+
1.7 Ondersteuning van oneindige sessies (MUST)[2]	%	%	%	+	+	+
1.8 Ondersteuning van ondeelbare events (MUST)[2]	%	+	%	+	+	+

1 Het beoordelen van de hoeveelheid overhead is relatief. Hier wordt naar de hoeveelheid overhead in het gehele protocol gekeken, in vergelijking tot een protocol als RADIUS.

requirement	***RADIUS***	***DIAMETER***	***TACACS+***	***SNMP***	***MSIX***	***ADIF***
1.9 Inter−domain accounting (MUST)	+	+	+	−	+	n
1.10 Meerdere accounting servers (MUST)	+	+		+	+	n
1.11 Samengestelde diensten (SHOULD)	−	−	−	−	+	n
2. Security requirements						
2.1 Integrity protection (MUST)	&	+	&	&	&	n
2.2 Authenticatie (MUST)	&	+	&	&	&	n
2.3 Confidentiality protection (MUST)	−	+	&	&	&	n
2.4 Replay protection (MUST)	−	+			+	n
2.5 Non−repudiation (SHOULD)	−	+	−	−	−	n
2.6 Brokers (MUST)	−	+	−	−	−	n
3. Accounting event requirements						
3.1 Start of a session (start bericht) (MUST)	+	+	+		+	n
3.2 End of a session (stop bericht) (MUST)	+	+	+		+	n
3.3 Update of a session (interim bericht) (MUST)	+	+	+		+	n
3.4 Session record (MUST)	%	%	%		+	+
3.5 Polling (MUST)	−	%	−	+	−	n
3.6 Event−driven polling (MUST)	−	−	−	+	−	n
3.7 Bevestiging van bericht (MUST)	+	+	+		+	n
3.8 Negotiation of transfer method and format capabilities (MUST)	−	−	−		+[3]	n
4. Transport requirements						
4.1 Betrouwbaar transport (MUST)	%	+	+	+	+	n
4.2 Ondersteuning grote berichten (MUST)	−	+	+		+	n
4.3 Snel van server veranderen (MUST)	+	+	−		−	n
4.4 Buffering van accounting gegevens (MUST)	+	+	+	+	+	n
4.5 Bidirectionele communicatie (MUST)	−	+	−		−	n
4.6 Flow control (MUST)	−	+	+		+	n
5. Record format requirements						
5.1 Tagged and typed data (MUST)	+	+	+	+	+	+
5.2 Standaard datatypen (MUST)	+	+	+	+	+	−
5.3 Extensible (MUST)	−	+	−	+	+	+

2 Accounting berichten bij oneindige sessies en ondeelbare events zijn te simuleren door het gebruik van berichten voor eindige sessies.

3 Met MSIX kunnen ondersteunde versies van de server opgevraagd worden. Er kan niet onderhandeld worden over het te gebruiken transportmechanisme of de beveiliging.

4 In DIAMETER kunnen attributen gegroepeerd worden op een enkel niveau. Er is geen nesting mogelijk.

requirement	*RADIUS*	*DIAMETER*	*TACACS+*	*SNMP*	*MSIX*	*ADIF*
5.4 Gegroepeerde of gestructureerde attributen (MAY)	–	&[4]	–	+	–	–
5.5 Human readable (MAY)	–	–	–	–	+	+
5.6 Compact record format (SHOULD)	+	+	+	+	–	+
5.7 Uitbreidbare berichten (MUST)	–	+	–	+	+	n
5.8 Verschillende diensten (MUST)	–	&	–	+	+	+
5.9 Definitie diensten (SHOULD)	–	–	–	+	+	+
5.10 Samengestelde diensten (MUST)	–	–	–	–	+	n

Tabel 6.1: vergelijking van accounting protocollen.

– *voldoet niet*
\+ *voldoet wel*
& *voldoet gedeeltelijk*
n *niet van toepassing*
% *bevat geen specifieke ondersteuning, maar is mogelijk*

7. Conclusies en aanbevelingen

Zoals eerder aangegeven is er behoefte aan een accounting protocol dat algemeen toepasbaar is. RADIUS is het standaard protocol dat wordt gebruikt bij inbelfaciliteiten. Dit protocol schiet echter tekort bij de huidige omgevingen waarbij ook sprake is van inter−domain accounting. Naast inbellen zijn er ook allerlei andere vormen van electronische dienstverlening waarbij accounting moet kunnen worden toegepast.

Om in de koppeling en uitwisseling van accounting gegevens tussen verschillende organisaties te kunnen voorzien is een standaard accounting protocol nodig. De requirements aan een dergelijk accounting protocol liggen wat betreft beveiliging, snelheid en betrouwbaarheid voor de verschillende toepassingen nogal eens ver uiteen. Daarom heeft een aanpak waarbij een verzameling van verschillende deelprotocollen wordt gebruikt de voorkeur [ietf−aaa−acct]. Deze opdeling zou moeten bestaan uit een transport protocol, een beschrijving van de uit te wisselen berichten en een record format. Deze zouden in een framework moeten worden gecombineerd.

In dit document zijn de requirements voor een algemeen accounting protocol volgens deze opdeling opgesomd en toegelicht, zodat deze apart ontworpen kunnen worden. Het is zo mogelijk om een accounting protocol samen te stellen uit de verschillende onderdelen, afhankelijk van de toepassing. Zo kunnen bijvoorbeeld andere deelprotocollen gebruikt worden voor real−time inter−domain usage−sensitive billing dan voor offline intra−domain trend analyse.

De requirements leiden nog niet vanzelf naar een accounting protocol. Hiervoor zijn ze nog te algemeen. Ze zijn bruikbaar om bestaande accounting protocollen te evalueren. Een nieuw protocol zou deze requirements kunnen gebruiken om specifieke eisen op te stellen.

Momenteel is er nog geen enkel protocol dat algemeen toepasbaar is. Wel zijn er initiatieven om goede accounting protocollen te ontwikkelen voor specifieke toepassingen. Zo is DIAMETER prima geschikt om RADIUS te vervangen. Deze protocollen combineren echter transport en record format in één protocol. In een nieuw protocol zouden transport en record format echter moeten worden gescheiden.

Met DIAMETER is een prima transport protocol op basis van UDP gedefinieerd. Het gebruik van TCP is echter efficiënter voor grotere hoeveelheden gegevens. Voor real−time accounting schiet TCP te kort door de slow start en gebruikte timeout parameters. Als het transport protocol van DIAMETER apart gespecificeerd zou worden, zou dit samen met TCP een goede verzameling van transportprotocollen vormen voor gebruik in accounting toepassingen.

Er kunnen verschillende record formats worden gebruikt. Hierbij kan worden gedacht aan een binair formaat zoals bij DIAMETER of een formaat gebaseerd op ASN.1. Deze formaten zijn vaak snel te genereren door eenvoudige hardware. Het is echter ook mogelijk om dezelfde informatie in een human−readable formaat als MIME of XML op te slaan. XML is flexibel en er kunnen ingewikkelde hyrachische structuren gedefinieerd mee worden. Het nadeel van XML is dat er voor het opslaan van relatief eenvoudige gegevens veel karakters nodig zijn. Het gebruik van compressie is dan aan te bevelen. Het gebruik van MIME, of iets soortgelijks, lijkt daarom voor accounting toepassingen meer voor de hand te liggen. Hierin is ADIF een interessante ontwikkeling.

MSIX voorziet als enige protocol in het definiëren van diensten. Het kunnen definiëren van eigen diensten, zonder hiervoor een lange standaardisatieweg te volgen, is belangrijk. Er zouden enkele basisdiensten kunnen worden gestandaardiseerd. Het zou dan mogelijk moeten zijn om met afleidingen van deze diensten nieuwe diensten te definiëren. Het is belangrijk dat deze nieuwe diensten in een bestaande accounting server kunnen worden geïntegreerd, zodat minimale aanpassingen nodig zijn bij het aanbieden van de nieuwe dienst.

Als verzameling berichten is in de requirements een verzameling genoemd. Een accounting protocol moet echter uitbreidbaar zijn met nieuwe soorten berichten.

Er is nog geen algemeen accounting protocol, maar een verzameling tools gebaseerd op delen van andere protocollen zoals het transport van DIAMETER, de dienstendefinitie van MSIX en de compactheid en leesbaarheid van ADIF lijkt een goede weg naar een algemeen toepasbaar accounting protocol.

Het definiëren van een framework waarin verschillende transportprotocollen en record formats gehangen kunnen worden is waarschijnlijk de beste oplossing. Zo kan afhankelijk van de toepassing de juiste verzameling deelprotocollen gebruikt worden. Bij eenvoudige diensten waarbij real−time accounting belangrijk is zouden DIAMETER achtige protocollen gebruikt kunnen worden, terwijl bij wat complexere diensten waarbij meer details over sessies wordt weergegeven een TCP verbinding en een XML record format voldoet.

Literatuurlijst

[arkko−acctreq] J. Arkko, "Requirements for Internet−Scale Accounting Management", Internet draft (work in progress), draft−arkko−acctreq−00.txt, August 1998.

[arkko−acctrqlis] A. Arkko, "Accounting Requirements", Internet draft (work in progress), draft−arkko−acctrqlis−00.txt, October 1999.

[blount−acct−msix] A. Blount, D. Young, "Metered Service Information eXchange Protocol Specification version 1.2", Internet draft (work in progress), draft−blount−acct−msix−00.txt, July 1999.

[blount−acct−service] A. Blount, "Accounting Protocol and Record Format Features", Internet draft (work in progress), draft−blount−acct−service−00.txt, September 1999.

[grant−tacacs] D. Carrel, L. Grant, "The TACACS+ Protocol Version 1.78", Internet draft (work in progress), draft−grant−tacacs−02.txt, January 1997.

[wang−aaa−cel−req] J. Wang, R. Wang, "Cellular Network Authentication, Authorization, adn Accounting Requirements", Internet draft (work in progress), draft−wang−aaa−cel−req−00.txt, October 1999.

[ekstein−nasreq−protocomp] R. Ekstein, Y. T'Joens, B. Sales, O. Paridaens, "AAA Protocols: Comparison between RADIUS, DIAMETER and COPS.", Internet draft (work in progress), draft−ekstein−nasreq−protocomp−01.txt, January 2000.

[calhoun−diameter−framework] P. R. Calhoun, G. Zorn, P. Pan, H. Akhtar, "DIAMETER Framework Document", Internet draft (work in progress), draft−calhoun−diameter−framework−05.txt, December 1999.

[calhoun−diameter] P.R. Calhoun, A.C. Rubens, "DIAMETER Base Protocol", Internet draft (work in progress), draft−calhoun−diameter−12.txt, December 1999.

[calhoun−diameter−accounting] J. Arkko, P.R. Calhoun, P. Patel, G. Zorn, "DIAMETER Accounting Extension", Internet draft (work in progress), draft−calhoun−diameter−accounting−03.txt, December 1999.

[ietf−aaa−acct] B. Aboba, J. Arkko, D. Harrington, "Introduction to Accounting Management", Internet draft (work in progress), draft−aboba−acct−02.txt, October 1999.

[ietf−aaa−na−reqts] M. Beadles et al., "Network Access AAA Evaluation Criteria", nternet draft (work in progress), draft−ietf−aaa−na−reqts−01.txt,October 1999.

[ietf−mobileip−aaa−reqs] S. Glass, S. Jacobs, C. Perkins, "Mobile IP Authentication, Authorization, and Accounting Requirements", Internet draft (work in progress), draft−ietf−mobileip−aaa−reqs−00.txt, October 1999.

[ietf−nasreq−criteria] M. Beadles, "Criteria for Evaluating Network Access Server Protocols", Internet draft (work in progress), draft−ietf−nasreq−criteria−03.txt, October 1999.

[ietf−radius−accounting−v2] C. Rigney, "RADIUS Accounting", Internet draft (work in progress), draft−ietf−radius−accounting−v2−01.txt, May 1999.

[ietf−radius−acct−interim] P. R. Calhoun, M. Beadles, A. Ratcliff, "RADIUS Accounting Interim Accounting Record Extension", Internet draft (work in progress), draft−ietf−radius−acct−interim−01.txt, January 1998.

[ietf−roamops−actng] B. Aboba, D. Lidyard, "The Accounting Data Interchange Format (ADIF)", Internet draft (work in progress), draft−ietf−roamops−actng−06.txt, August 1999.

[ietf−rap−cops] J. Boyle, R. Cohen, D. Durham, S. Herzog, R. Rajan, A. Sastry, "The COPS (Common Open Policy Service) Protocol", Internet draft (work in progress), draft−ietf−rap−cops−08.txt, November 1999.

[rfc1272] C. Mills, D. Hirsh, G. Ruth, "Internet Accounting: Background", RFC 1272, November 1991.

[rfc1521] N. Borenstein, N. Freed, "MIME (multipurpose Internet Mail Extensions) Part One: Machanisms for Specifying and Describing the Format of Internet Message Bodies", RFC 1521, December 1993.

[rfc2138] C. Rigney, A. Rubens, W. Simpson, S. Willens, "Remote Authentication Dial In User Service (RADIUS)", RFC 2138, January 1997.

[rfc2139] C. Rigney, "RADIUS Accounting", RFC 2139, April 1997.

[rfc2194] B. Aboba, J. Lu, J. Alsop, J. Ding, W. Wang, "Review of Roaming Implementations", RFC 2194, September 1997.

[rfc2477] B. Aboba, G. Zorn, "Criteria for Evaluating Roaming Protocols", RFC 2477, January 1999.

[rfc2607] B. Aboba, J. Vollbrecht, "Proxy Chaining and Policy Implementation in Roaming", RFC 2607, June 1999.

[rfc2205] E. R. Braden, L. Zhang, S. Berson, S. Herzog, S. Jamin, "Resource ReSerVation Protocol (RSVP) −− Version 1 Functional Specification", RFC 2205, September 1997.

[msixwww] MSIX (Metered Service Information eXchange) homepage, http://www.msix.org/

[xmlw3c] T. Bray, J. Paoli, C. Sperberg−McQueen, "Extensible Markup Language (XML) 1.0", W3C Recommendation, February 1998.

[billaudit] Infozech: Telephone Bill Audit and Overcharge Recovery Service, http://www.infozech.com/audit.html

[ts101321] "Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON); Inter−domain pricing, authorization, and usage exchange", TS 101 321 V1.4.2, December 1998.

[ofxwww] OFX: Home Page, http://www.ofx.net/

[etsiwww] ETSI (European Telecommunications Standards Institute) Home Page, http://www.etsi.org/

[rsvpwww] RSVP Project, http://www.isi.edu/div7/rsvp/

[rsvpacc] Accounting model based on integrated RSVP/intserv and diffserv architecture, http://ing.ctit.utwente.nl/WU5/ongoing/qosmodel/qos−model.html

[snmpintro] "An Introductory Overviw of SNMP", Diversified Data Resources Inc., http://www.ddri.com/, 1999.

[thoughtsmail] "Some thoughts on data representation", Email to AAAarch Reseach Group <aaaarch@fokus.gmd.de> from David W. Spence <dwspence@merit.edu>, mail archive: http://www.fokus.gmd.de/research/cc/glone/research/aaaarch/

[aaaterms] A. de Jong, "AAA Frequently Used Terms", http://ch.twi.tudelft.nl/~arthur/aaa/aaaterms.html