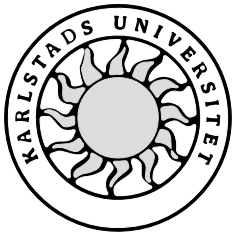


Datavetenskap

Johannes Larsson och John Langered

Virtuellt privat nätverk: Utredning av begrepp med konfigurationsexempel

---

Examensarbete, C-nivå

2003:13

Virtuellt privat nätverk: Utredning av begrepp med konfigurationsexempel

Johannes Larsson och John Langered

© 2003 Författarna och Karlstads universitet

Denna rapport är skriven som en del av det arbete som krävs för att erhålla en kandidatexamen i datavetenskap. Allt material i denna rapport, vilket inte är mitt eget, har blivit tydligt identifierat och inget material är inkluderat som tidigare använts för erhållande av annan examen.

Johannes Larsson

John Langered

Godkänd, 2003-06-03

Handledare: Thijs Holleboom

Handledare Karlstad kommun, IT-enheten:
Ronny Larsson

Examinator: Stefan Lindskog

Sammanfattning

Karlstad kommun vill ha ett säkert sätt ge sina anställda åtkomst till kommunens intranät via Internet. Syftet med denna rapport är att behandla en möjlig lösning som kallas *Virtuellt Privat Nätverk* (VPN).

Rapporten utreder begrepp inom VPN samt innehåller en guide för ett par olika konfigurationer med kommunens material. Vi tar upp begrepp och definitioner som gäller inom området, samt i vissa fall förklaras en del av de bakomliggande datasäkerhetsmekanismerna.

För detta ändamål har vi av kommunen fått viss bestämd hård- och mjukvara, bestående av en Cisco VPN Concentrator 3005, Cisco VPN Client samt en Cisco 1600 Series Router. VPN Concentrator 3005 är en specialanpassad VPN-server, VPN Client är en mjukvara för uppkoppling mot koncentratorn och routern används för att prova hur adressöversättning fungerar i samband med VPN, då detta kan orsaka problem i vissa fall.

Vi tar upp två VPN-konfigurationer med olika säkerhetsnivåer och mäter hur mycket de belastar uppkopplingen.

De tester som vi genomfört har fungerat väl i testmiljön och om det inte uppstår andra komplikationer i en ”verklig” miljö verkar VPN vara en mycket bra lösning för Karlstad kommun.

Virtual Private Network: Concepts and configuration examples

Abstract

The municipality of Karlstad wants a secure way for its employees to access the municipal intranet via the Internet. The purpose of this report is to discuss a possible solution called *Virtual Private Network* (VPN).

This report investigates the concepts within VPN for the municipality of Karlstad and contains a guide for a couple of different configurations. We look at some concepts and definitions that apply to this area, and in some cases explain the underlying data security mechanisms.

We have specific hard- and software for this purpose, consisting of a Cisco VPN concentrator 3005, the Cisco VPN Client software and finally a Cisco 1600 Series Router. The concentrator is a special-purpose server for VPN traffic, and the client is a special-made software that handles the connections to the concentrator. The only purpose of the router is to test network address translation in combination with VPN, which is known to cause problems.

There are two configurations described within this report that uses different levels of security. We measure the time that it takes to transfer files with different sizes with these configurations and compare them to a configuration that uses no encryption.

The tests conducted in the laboratory environment turned out well and if no other complications arise in a ”real” environment, VPN seems to be a good solution for the municipality of Karlstad.

Innehållsförteckning

Figurförteckning

Tabellförteckning

1 Inledning

Karlstad kommun har i dagsläget inget säkert sätt för anställda att arbeta med datorbaserade verktyg från andra platser än kommunens lokaler. Det gör att om den anställde vill fortsätta arbeta med ett dokument hemma måste denne skicka det via e-post till hemmet, eller kopiera det till diskett eller annat flyttbart medium. Det förstnämnda innebär en klar säkerhetsrisk, då detta oftast sker på ett helt okrypterat sätt över ett osäkert medium, Internet. Det andra alternativet blir mindre och mindre användbart då det mest utbredda, flyttbara mediet är disketter. Dessa har en liten lagringskapacitet och kombineras detta med det ständigt växande informationsbehovet uppstår problem. Därutöver finns det även behov att arbeta direkt mot lokala databaser och program på Karlstad kommun, vilket är omöjligt med de två metoderna beskrivna ovan. En teknik som löser problemet kallas *Virtual Private Network* (VPN).

I rapporten behandlas olika aspekter, begrepp och varianter av virtuella privata nätverk, så kallade VPN*s*. Utöver detta redovisas även ett prestandatest för tre olika VPN-konfigurationer.

Kapitel två tar upp de grundläggande begreppen inom datasäkerhet som är av betydelse för att förstå VPN-tekniken. Här beskrivs fem viktiga grundpelare inom området: kryptering, integritet, autentisering, tillgänglighet och utbildade användare.

Kapitel tre behandlar VPN mer specifikt och då främst de protokoll som används för att överföra informationen. Utöver detta ges en beskrivning över den generella strukturen för ett VPN. Här redovisas även resultatet av prestandatestet.

Det fjärde kapitlet är en steg-för-steg-anvisning för två olika konfigurationer med Ciscos VPN-lösning. Hårdvaran består av en *Cisco Concentrator 3005* och mjukvaran av *Windows 2000 Advanced server* för certifikathantering samt Ciscos klientmjukvara.

2 Datasäkerhet allmänt

Datasäkerhet har blivit mer och mer viktigt allteftersom användandet av Internet har spridit sig och lättanvänd mjukvara avsedd för personer med oärliga avsikter är enkelt att få tag på. Detta har inneburit att marknaden för produkter som skyddar mot ”crackers” har blivit väldigt stor och därför även utbudet på lösningar. I denna rapport avses med begreppet datasäkerhet kryptering, integritet, autentisering, tillgänglighet och utbildade användare.

2.1 Kryptering

När hemlig data ska skickas över ett publikt nät är det inte lämpligt att detta sker i klartext. Detta löses genom att kryptera meddelandet, vilket innebär att det är omöjligt att tyda det verkliga innehållet utan att först dekryptera. Krypteringen är indelad i två huvudgrupper, symmetriskt och asymmetrisk kryptering. För båda grupperna sker krypteringen genom att en krypteringsalgoritm använder sig av en nyckel för att skapa ett chiffer. Det är inte algoritmen i sig som är hemligheten bakom chiffret utan nyckeln som den använder sig av.

Det kanske enklaste exemplet på en krypteringsalgoritm är ett så kallat Caesarchiffer vilket innebär att varje bokstav i ursprungsmeddelandet byts ut mot en annan som ligger exakt n positioner ifrån den i ett givet alfabet. I detta fall är det talet n som är nyckeln och om meningen ”Kryptering är bra” krypteras med nyckeln 3 erhålles ”Nuäsxhulqj bu eud”. Notera att om sista bokstaven passeras fortsätter positionsstegningen från början av alfabetet. För exemplet ovan innebar det att ”ä” substituerades mot ”b”.

För att uppnå olika säkerhetsgrader på chiffret används olika längder på nycklarna. Generellt gäller att ju längre nyckel desto högre säkerhet för algoritmen. Dock innebär en längre nyckel även mer krävande beräkningar vilket påverkar prestanda. Nyckellängden anges av antalet bitar. Vanligt längd är 56 till 1024 bitar.

2.1.1 Symmetrisk kryptering

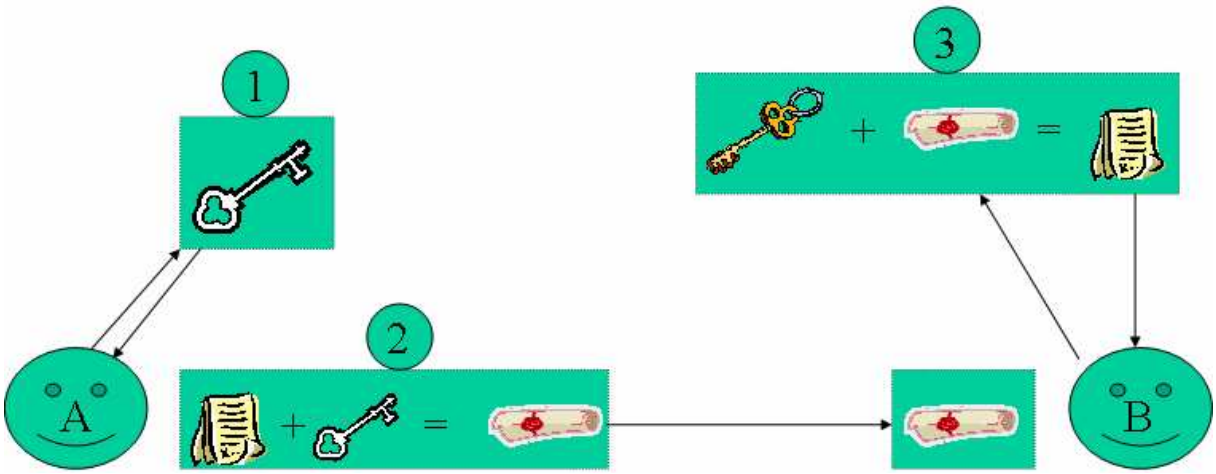
Symmetrisk kryptering innebär att data krypteras och dekrypteras med samma nyckel, en så kallad ”shared secret”-nyckel. Detta medför att alla inblandade måste känna till nyckeln, vilket resulterar i en säkerhetsrisk som uppkommer i och med distribueringen av nyckeln. Fler inblandade innebär även en större sannolikhet att någon obehörig kommer över nyckeln på grund av användarnas oförsiktighet och/eller okunnighet. Ytterligare en nackdel med

symmetrisk kryptering är att om nyckeln skulle komma i någon obehörigs händer måste samtliga användare byta nyckel, eftersom erövraren av nyckeln annars kan läsa den information som skickas krypterad. I det asymmetriska fallet räcker det med att den användaren vars nyckel blivit avslöjad utför nyckelbytet.

2.1.2 Asymmetrisk kryptering

”Privat/publik nyckel”-lösningen [2] är ett exempel på asymmetrisk kryptering. Varje användare härleder ett nyckelpar från två stora primtal. Den ena nyckeln benämns som privat och den andra som publik. Den privata får under inga omständigheter offentliggöras medan den publika delas ut till alla som vill kommunicera säkert med dess ägare. Gällande för nyckelparet är att det som krypteras med den ena nyckeln endast kan dekrypteras med den andra och vice versa. En förutsättning är att den ena nyckeln ej kan härledas från den andra. Om en klient A vill skicka ett krypterat meddelande till en klient B sker följande enligt figur nedan:

1. A erhåller Bs publika nyckel via exempelvis Bs hemsida.
2. Sedan använder A denna nyckel till att kryptera meddelandet den önskar skicka.
3. När B tar emot meddelandet använder den sin privata nyckel för att dekryptera det.



Figur 2.1: Exempel på överföring med asymmetrisk kryptering

Om en mottagare får ett meddelande som är krypterat med en avsändares privata nyckel kan den dekryptera det med den publika nyckeln och på så sätt vara säker på att det är rätt avsändare som har skickat meddelandet. Skulle någon obehörig komma över avsändarens nyckel innebär det att denne kan skicka meddelanden i den ursprungliga avsändarens namn och läsa meddelanden krypterade med dennes publika nyckel. Skadan är alltså inte lika omfattande som vid stöld av nyckeln vid symmetrisk kryptering. Det är viktigt att alla kommunikationsparter informeras om detta (när detta upptäcks) och ett nytt nyckelpar skapas.

Dock räcker det med att den drabbade genomför nyckelbyte, övriga kan behålla sina nyckelpar.

Då den asymmetriska krypteringen innebär mer krävande beräkningar än den symmetriska lösningen är det vanligt att asymmetrisk kryptering används endast vid autentiseringen (se 2.3 Autentisering). När detta är gjort och en säker uppkoppling har skapats mellan parterna förhandlas en symmetrisk nyckel fram, som är unik för sessionen. Denna nyckel används för att kryptera och dekryptera informationsutbytet.

2.2 Integritet

När data skickas vill varken sändaren eller mottagaren att den skickade datan blir manipulerad på vägen till sin destination. Om till exempel en begäran på en transaktion skickas till banken kan det vara frestande för en ”cracker” att ändra mottagarkonto och/eller belopp. Detta undviks genom att ett tal härleds, som bygger på och är unikt, för det data som ska skickas, med hjälp av någon algoritm. Det kan ses som att ett digitalt fingeravtryck appliceras på datan, och den metod som används kallas ”hashing”. De vanligaste formerna av ”hashing” är *Message Digest 5* (MD5) och *Secure Hash Algorithm* (SHA). Dessa algoritmer skapar ett hashvärde av bestämd längd, som för SHA är 60 bitar. Det är i princip omöjligt att återskapa det ursprungliga meddelandet från hashvärdet, och eftersom det endast tar 60 bitar i anspråk är det ett billigt sätt att försäkra sig om att informationen inte ändrats av någon illvillig person (se 2.3.1 Digitala signaturer).

2.3 Autentisering

När kommunikation upprättas mellan en server och en klient är det ofta av största vikt att båda vet den andres identitet. För att fortsätta på bankexemplet: För att en transaktion ska godkännas måste banken veta vem det är som vill få den utförd. Det skulle vara katastrof om någon obehörig fritt kunde disponera någon annans konto. Banken vill vara helt säker på att det är rätt person som gör transaktionen, vilket kan ske på flera olika sätt.

Det finns ett antal olika metoder för autentisering som är indelade i svaga och starka. De svaga ger i informationskänsliga sammanhang inte tillräckligt hög säkerhet då lösenordet skickas i klartext och lätt kan fångas upp av obehöriga. De starka skickar lösenord och annan autentiseringsinformation krypterat och ger då hög säkerhet. Ett exempel på svag autentisering är användarnamn och lösenord, som skickas okrypterat över Internet. Denna metod erbjuder endast grundläggande säkerhet. Stark autentisering kan genomföras med till

exempel digitala certifikat. Dessa bygger på tillit, det vill säga parterna måste lita på certifikatutgivaren. Certifikat erhålles via en *Certificate Authority* (CA), till exempel Verisign inc., eller genom att organisationen blir sin egen CA. En CA är ett företag som går i god för att certifikatet som används är äkta. Att tänka på vid införskaffande av certifikat är vilket som är lämpligast av så kallade mjuka eller hårda certifikat [10]. Som namnen antyder är mjuka certifikat filbaserade och kan kopieras och skickas över Internet medan hårda certifikat är fysiska objekt så som ett kort eller dylikt. Vilken av lösningarna som är aktuell för användaren beror dels på vilken säkerhetsnivå som är av intresse och dels vilka resurser som finns att tillgå. Vad det gäller säkerheten så är ett hårt certifikat svårare att göra kopior av och anses därför säkrare. Dock innebär de hårda certifikaten en dyrare lösning än de mjuka. Till att börja med måste det finnas någon enhet på datorn som kan tillgodogöra sig informationen från det fysiska objekt som certifikatet ligger sparat på, till exempel en kortläsare. Dessa har en tendens att krångla och behov att teknisk support uppstår. Framåt i tiden förutspås kortläsare vara standard i PCn och då underlättas användandet av hårda certifikat. Gällande distributionen är det en mer omständig process att få ut de fysiska objekten till varje enskild användare än i fallet med mjuka då det går att ladda ner certifikaten från en hemsida på Internet.

För att skapa en förbindelse med certifikat krävs det att varje användare har två certifikat installerade. Det ena är ett så kallat rotcertifikat vars syfte är att påvisa tillit för certifikatutfärdaren för de certifikat som ska autentiseras. Det andra är ett personligt certifikat som identifierar användaren (binder den publika nyckeln till användarens identitet). För att kommunicera med till exempel en användare som har ett Verisign-certifikat krävs det att det finns ett rotcertifikat från Verisign installerat.

2.3.1 Digitala signaturer

En digital signatur [3] används för att säkerställa avsändaren av ett meddelande, till exempel e-post eller en bildfil. För att skapa en digital signatur krävs det ett publikt/privat-nyckelpar och ett digitalt certifikat. Proceduren fungerar enligt följande princip:

1. Sändarens meddelande hashas med en hashalgoritm, till exempel SHA eller MD5, och resultatet blir en så kallad “message digest” (se 2.2 Integritet).
2. Det ursprungliga meddelandet kan svårligen återskapas från det hashade meddelandet men det är inte svårt att skapa en ny ”message digest” för en obehörig som kommer över det skickade meddelandet. Därför används sändarens privata nyckel för att kryptera det hashade meddelande så mottagaren kan autentisera det.

3. Det hashade meddelandet skickas med det ursprungliga e-postmeddelandet och blir sändarens signatur.

På mottagarsidan sker följande:

1. Sändarens publika nyckel används för att dekryptera den digitala signaturen.
2. För att säkerställa att meddelandet är intakt så gör mottagaren en hash på klartext-meddelandet.
3. Om det hashade meddelandet överensstämmer är mottagaren garanterad att meddelandet är oförändrat och att avsändaren är rätt.

Observera att meddelandet i sig inte behöver krypteras när en digital signatur skapas.

2.4 Tillgänglighet

Att tänka på gällande säkerheten i ett system är att den inte får inskränka på systemets tillgänglighet mer än nödvändigt. Det gäller att göra en avvägning mellan hur hög säkerhet som önskas respektive hur lätt det ska gå att komma åt önskad information. Om det tar en timme att upprätta en VPN-förbindelse, på grund av olika säkerhetsmekanismer, mellan en klient och kommunens intranät så är det inte en acceptabel lösning trots att säkerhetskraven mer än väl uppfylls. Den låga tillgängligheten kommer med stor sannolikhet göra att VPN-resursen inte utnyttjas i någon större utsträckning av personalen.

Även överföringshastigheten påverkas av den nivå av säkerhet som väljs. Organisationen som implementerar ett VPN måste ta ställning till hur stark kryptering som ska användas, då denna i stor utsträckning kommer påverka överföringshastigheten. Ju större nyckel som används, desto större blir den overhead-information som måste skickas med varje paket. Som jämförelse mellan val av krypteringsalgoritm samt nyckellängd och överföringshastighet (se 3.5 Prestandamätning).

2.5 Utbildade användare

Ett förhållandevis vanligt sätt att ta reda på användarnamn och lösenord är ”social engineering”, vilket innebär att användare luras att frivilligt ge ifrån sig dessa uppgifter. En ”cracker” kan till exempel ringa upp användaren och utge sig för att vara administratör och fråga efter vederbörandes inloggningsuppgifter. I alldeles för många fall ger den outbildade användaren då ut sina uppgifter i god tro. Det spelar i sådana fall nästan ingen roll hur

kraftfulla säkerhetsmekanismer som används i övrigt om ”crackern” enkelt kan lura till sig åtkomst av informationen.

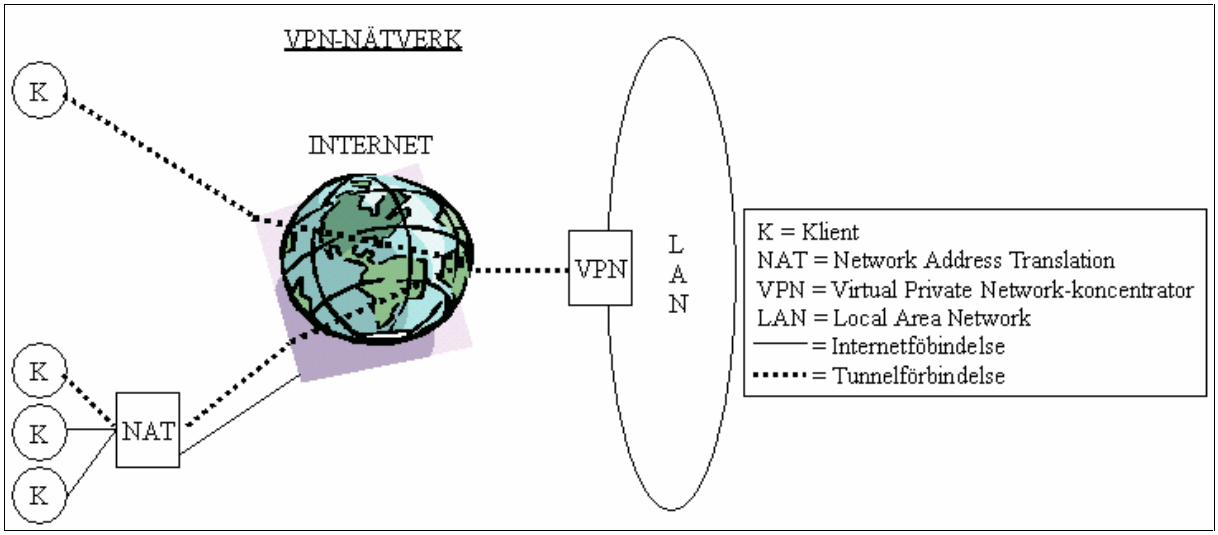
3 VPN – översikt

Föregående kapitel berör ett flertal begrepp inom datasäkerhet på ett översiktligt sätt. Detta kapitel ger en beskrivning av vad ett *Virtuellt Privat Nätverk* (VPN) är samt hur begreppen inom datasäkerhet tillämpas för att göra ett VPN säkert.

3.1 Vad är ett VPN?

Ett VPN är ett sätt att säkert skicka information över ett annars osäkert medium. Med säkert menas i den här rapporten att ingen obehörig kan läsa eller förändra informationen på vägen. Detta medium är nästan alltid Internet, men det förekommer även inom *Local Area Networks* (LAN). Problemet med att skicka klartextinformation över Internet är att någon med rätt utrustning och kunskap kan ”avlyssna” trafiken, det vill säga se vad som sänds. Oftast spelar detta ingen större roll, men för ett företag som handskas med sekretessbelagd och/eller känslig information innebär detta ett stort problem.

Genom att använda VPN kan en tunnelförbindelse skapas till ett privat nätverk [4] över ett osäkert nätverk. Detta innebär i praktiken att det räcker för klienten att ha tillgång till Internet för att komma åt det privata nätverket på ett säkert sätt, och användaren märker ingen skillnad mellan att sitta fysiskt ansluten och att vara ansluten över ett VPN. Denna lösning är betydligt billigare än alternativet att dra en direktförbindelse från alla klienter som vill komma åt nätverket.



Figur 3.1: Översikt VPN

Bilden ovan visar översiktligt hur en tunnelförbindelse kan vara skapad mellan en klient som är direkt uppkopplad mot Internet och en klient som befinner sig i ett LAN där interna nätverksadresser används (NAT) mot en VPN-server (koncentrator)

3.2 De olika delarna i ett VPN

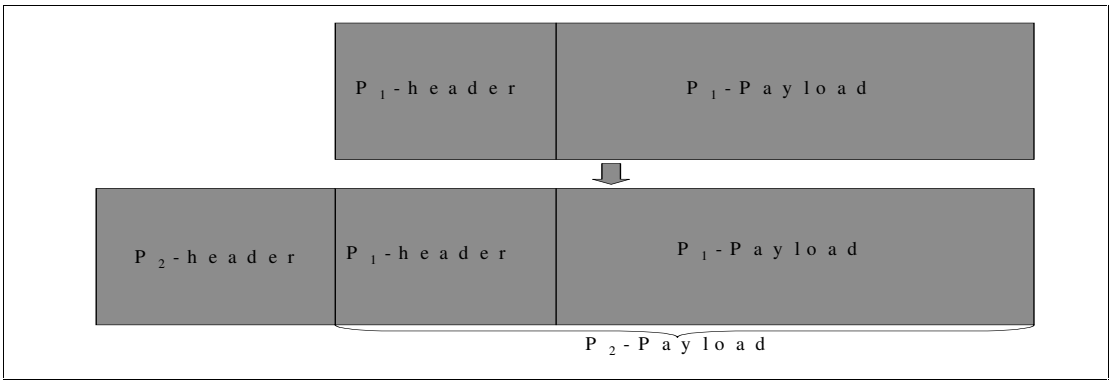
Alla VPN bygger på fyra grundstenar:

- Tunnling
- Autentisering
- Kryptering
- Behörighetskontroll

Inom varje grundsten finns det olika lösningar att tillgå, och en eller flera väljs efter företagets behov.

3.2.1 Tunnling

För att förklara tunnling beskriver vi här ett exempel där detta används:
Internt i nätverk A används protokoll P_1. Information ska utbytas med klient B som använder sig av samma protokoll P_1. A och B är sammankopplade via ett medium, till exempel Internet, som använder sig av ett annat protokoll P_2. Data från A kan då inte skickas i sin ursprungliga form då det mellanliggande mediet inte stödjer P_1. En lösning till detta är att låta informationen tillsammans med P_1-headern utgöra payload i ett P_2-paket, det vill säga ett P_1-paket ”kapslas in” i ett P_2-paket. Payload utgör den information som skickas ner från överliggande lager i OSI-modellen, och header är den information som protokollet lägger till.



Figur 3.2: Översiktbild av protokollinkapsling

Som standard går all trafik genom tunneln när denna har etablerats. Dock kan det ibland vara önskvärt att viss trafik går direkt ut på Internet eller till någon nod i det lokala nätverket. Då kan en funktion som kallas s*plit tunneling* användas. Med split tunneling kan viss trafik gå

direkt från klientdatorn till angiven mottagare. Detta har både för- och nackdelar. Fördelarna består i att användaren kan skriva ut på sin lokala nätverksskrivare och andra resurser på nätverket. Den största nackdelen är säkerhetsrisken som detta skapar. En angripare kan under pågående tunnelkommunikation ta över eller på annat sätt komma åt det nätverk som klienten kommunicerar med. De två metoderna som finns för att konfigurera split tunneling är att endera ange vilken trafik som ska tunnlas, eller att ange vilken som inte ska tunnlas.

3.2.2 Autentisering

Autentiseringdelen i ett VPN innebär att både server och klient säkerställer den andres identitet. Därefter kan datan börja sändas mellan dem. Ett sätt att hantera autentiseringen är att använda sig av digitala certifikat. Dessa certifikat är dokument som intygar att kommunikationsparten är den som den utger sig för att vara. Det är även viktigt att användaren vid klientdatorn har autentiserat sig vid inloggning med till exempel användarnamn och lösenord. Görs inte detta, och certifikatet lagras lokalt på hårddisken, kan vem som helst använda datorn och utge sig för att vara certifikatägaren.

3.2.3 Kryptering

Eftersom data mellan de kommunicerande parterna skickas över ett publikt nät, i detta fall Internet, är det inte säkert att skicka den i klartext då det med lätthet skulle kunna avlyssnas av personer med rätt kompetens och utrustning. För att erhålla säker kommunikation krypteras datadelen i IP-paketet med en krypteringsalgoritm. Ytterligare ett krav är att integritetskontroll utförs på data när den anländer till ändnoden. Detta för att vara säker på att informationen inte har manipulerats eller på annat sätt ändrats under överföringen.

Vid implementering av krypteringsalgoritmer bestäms först huruvida krypteringen ska ske tecken för tecken eller i block. Det förstnämnda kallas ”Streaming cipher” [8], där de största fördelarna är att det går mycket snabbare. Detta då det inte föreligger något behov av att vänta på hela block med data och felhanteringen blir inte lika omfattande. Anledningen till detta är att om det blir ett fel så påverkas endast en byte data. När den andra metoden används, så kallad ”Block cipher” [9], krypteras hela block med data av fixerad längd. Fördelarna med denna metod är i stället att krypteringen blir säkrare. Det krypterade meddelandet beror på flera tecken, eller bytes, och ett fenomen som kallas diffusion uppstår. Med detta menas att om ett tecken ändras påverkas krypteringen av efterföljande tecken.

3.2.4 Behörighetskontroll

Om en klient får tillträde till det privata nätverket bör åtkomsten av resurser begränsas beroende på vilka behov användaren har. Detta kan ske via en *Access Control List* (ACL) där åtkomsträttigheterna för användare sparas. I en ACL kan administratören tilldela användare tillgång till objekt som till exempel filer, mappar och hela hårddiskar.

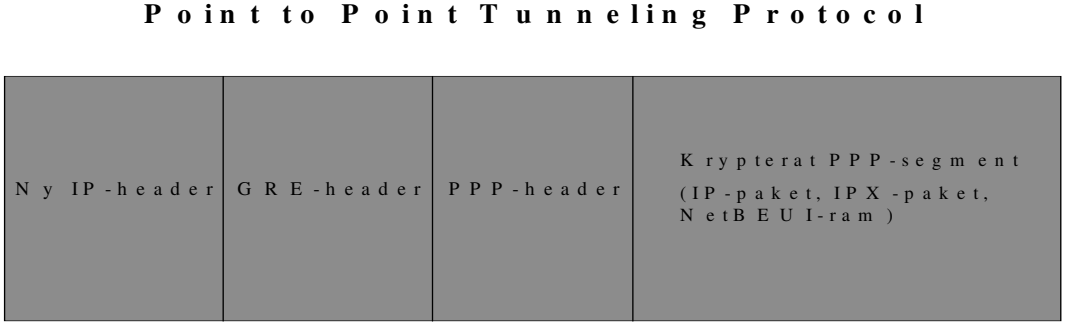
3.3 VPN-protokoll

För att skapa en tunnel krävs det att ett tunnelprotokoll används. Det finns en mängd av dessa men tre av de mest vanligt förekommande, som har störst sannolikhet att bli standardprotokoll, är *Point to Point Tunneling Protocol* (PPTP), *Layer 2 Tunneling Protocol* (L2TP) och *Internet Protocol Security* (IPsec). Det finns många olika faktorer som påverkar vilket protokoll som anses vara mest lämpligt för ett givet nätverk och varje protokoll har i sig ett antal möjliga konfigurationer och kombinationer.

Den främsta förespråkaren för PPTP är Microsoft och protokollet finns integrerat i de flesta av deras operativsystem. De fördelar som PPTP har gentemot de andra är att det är enklare att implementera och använda, dock är säkerheten inte lika hög. De andra två kan kombineras eller användas var för sig. Vi kommer här att koncentrera oss på IPsec.

3.3.1 PPTP

Point to Point Tunneling Protocol (PPTP) togs fram av Microsoft och US Robotics som ett sätt att upprätta säker kommunikation över osäkra medium. Idén är att packa in PPTP-datagram i IP-paket enligt bilden nedan och skicka över IP-nätverk, som Internet.



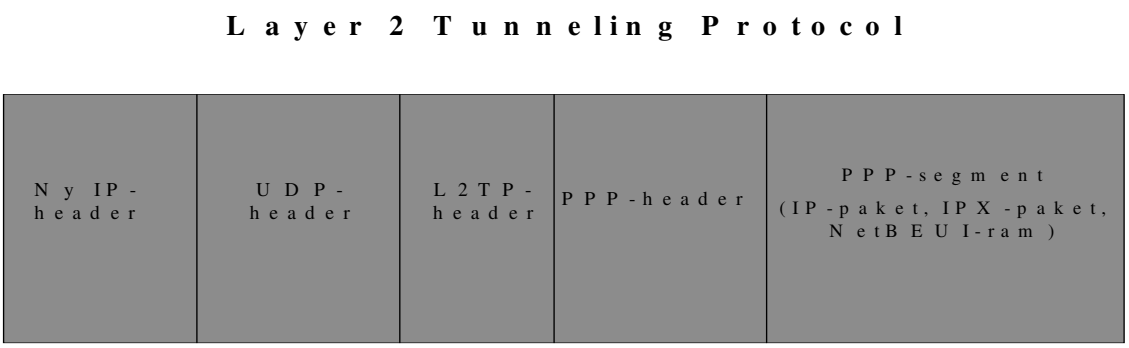
Figur 3.3: Point to Point Tunneling Protocol – datagrambeskrivning

PPTP-datagrammet är egentligen ett *Point to Point Protocol*-datagram (PPP) som krypterats och kapslats in i ett *Generic Routing Encapsulation*-datagram (GRE). GRE tas inte

upp i detta dokument då det är oväsentligt i sammanhanget. Ytterst i datagrammet placeras den nya IP-headern.

3.3.2 L2TP

L2TP [5] är en de facto-standard för VPN-tunnling. Principen bakom protokollet är att en PPP-ram läggs i en L2TP-ram enligt bilden nedan och sedan tunnlas denna över ett annat ”lager två”- eller ”lager tre”-nätverk.



Figur 3.4: Layer 2 Tunneling Protocol - datagrambeskrivning

L2TP är egentligen en kombination av *Layer 2 Forwarding* (L2F) och PPTP där de bästa egenskaperna valts ut. L2TPs funktion i ett VPN är att skapa och underhålla tunnelförbindelser. L2TP-protokollet har i sig inget stöd för kryptering men detta erhålles genom IPsec. Detta samarbete mellan protokollen är så vanligt att ”L2TP över IPsec” är ett begrepp. Det är tekniskt möjligt att använda L2TP utan kryptering men det rekommenderas inte eftersom informationen som skickas då ej är skyddad.

3.3.3 IPsec

Internet Protocol security (IPsec) utvecklades för säkra VPN-anslutningar av *Internet Engineering Task Force* (IETF). Innan IPsec fanns koncentrerades säkerhetslösningarna till applikationslagret, men detta innebar en del säkerhetsrisker då bland annat avsändar- och mottagaradress inte skyddades. Syftet med IPsec består i behovet av att kunna erbjuda kryptering, integritet, autentisering och skydd mot attacker av typen *Denial of Service* (DOS).

IPsec är inget eget protokoll, utan ett samlingsnamn för ett flertal protokoll som hanterar mekanismer för säker kommunikation på IP-lagret. De protokoll som tillsammans kallas IPsec är *Internet Key Exchange* (IKE), *Authentication Header* (AH) och *Encapsulation Security Payload* (ESP). De två sistnämnda finns i två versioner, *tunnel mode* och *transport mode* [13].

Då detta arbete inriktar sig på VPN baserat på enbart IPsec kommer vi här att beskriva protokollen i tunnel mode. I detta läge krypteras både IP-header och payload, till skillnad från transport mode som endast krypterar payload. Det sistnämnda är till för tunnlar som skapas med PPTP eller L2TP över IPsec.

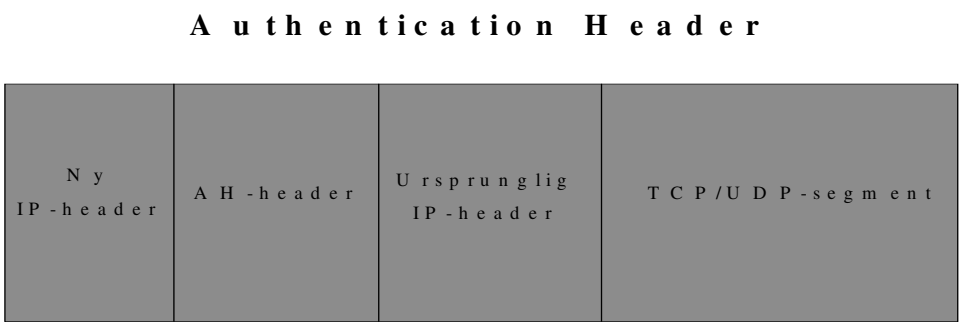
3.3.3.1 Internet Key Exchange

Internet Key Exchange (IKE) är ett förhandlingsprotokoll som hanterar autentisering av publika nycklar, genererar den information som behövs för att varje part ska kunna härleda en symmetrisk sessionsnyckel samt förhandlar fram alla säkerhetsparametrar som gäller för det aktuella kommunikationstillfället. Dessa säkerhetsparametrar behöver inte vara desamma från gång till annan, utan bestäms gemensamt av klient och server. Detta beror på vad de kommunicerande parterna kan hantera för säkerhetsinställningar.

För att kunna komma överens om en gemensam, symmetrisk sessionsnyckel över ett publikt nät använder sig IKE av ett protokoll som kallas för *Diffie-Hellman* [14].

3.3.3.2 Authentication Header

AH är ett av de två säkerhetsprotokollen som IPsec hanterar. Protokollet tillhandahåller integritet och autentisering [11].



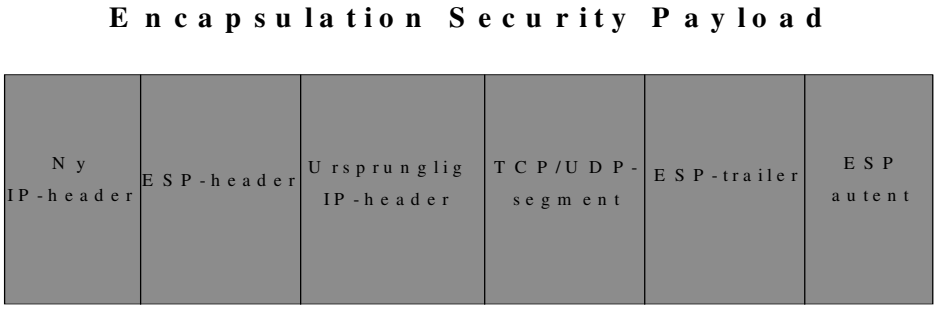
Figur 3.5: Authentication Header – datagrambeskrivning

I AH-headern ligger en message digest, skapad med en hash-algoritm, baserad på hela TCP/UDP-segmentet samt sändar- och mottagaradresserna. Det enda som inte tas med i hash-värdet är de fält som förändras på vägen, såsom ”hop count” och ”fragment offset”. Om något av de statiska värdena ändras kommer inte det hashade värdet att överensstämma med datagrammet, och mottagaren kan se att en förändring har skett. Det felaktiga paketet kastas därmed. Integriteten bevaras i och med detta, och som en positiv biprodukt fås även autentisering, då det hashade värdet måste krypteras. Om kryptering ej sker erhålles heller inte

integritet, då det är lika lätt för en inkräktare som en legitim användare att skapa ett hashvärde när hash-algoritmen är känd.

3.3.3.3 Encapsulation Security Payload

ESP är i praktiken allt som AH är, och lite till. ESP har således integritet och autentisering, men även konfidentialitet [12]. Ansatsen är dock lite annorlunda jämfört med AH, då ESP inte signerar den yttersta headern på paketet. Detta innebär att integriteten inte blir lika hög, men möjliggör i stället att ESP kan användas när paketen ska skickas från en klient som använder sig av *Network Address Translation* (NAT).



Figur 3.6: Encapsulation Security Payload - datagrambeskrivning

Precis som med AH skapas ett hashvärde, men här signeras inte autentiserings-headern och den nya IP-headern. Kryptering sker på den ursprungliga IP-headern, TCP/UDP-segmentet och ESP-trailern.

3.3.3.4 Kryptering med IPsec

Det finns tre krypteringsalgoritmer att välja bland för IPsec-kryptering, *Data Encryption Standard* (DES), *Triple DES* (3-DES) och *Advanced Encryption Standard* (AES).

DES [6] är en äldre algoritm utvecklad av IBM 1977 för kryptering men håller fortfarande en tillräckligt hög säkerhetsnivå för de flesta användningsområden. Den använder sig av en 56 bitar stark nyckel vid kryptering. Krypteringen sker i 18 steg. Det första steget är en permutation av meddelandet och sedan följer 16 stycken likartade beräkningar, där varje har resultatet av den föregående beräkningen som indata. Avslutningsvis sker ytterligare en permutation.

3-DES [1] är i grund och botten samma algoritm som DES med den skillnaden att DES-krypteringen upprepas tre gånger. Det finns två olika metoder att använda algoritmen, *Encryption-Encryption-Encryption* (EEE) och *Encryption-Decryption-Encryption* (EDE).

EEE-metoden fungerar så att resultatet från första krypteringen krypteras igen med en ny nyckel och vidare krypteras utdata därifrån med den tredje och sista nyckeln.

EDE har tre olika metoder för kryptering, och vilken som används beror på antalet unika nycklar. Om tre unika nycklar används sker först en kryptering med nyckel ett, följt av en dekyptering med nyckel två. Eftersom dekrypteringen sker med en annan nyckel än den för kryptering kommer resultatet att vara helt oläsligt. För att ytterligare bättra på säkerheten krypteras detta med ytterligare en nyckel. Metod två baserar sig på två unika nycklar. Metoden är likadan som för den med tre nycklar, det enda som skiljer är att nyckel tre är identisk med nyckel ett. Krypteringen sker med nyckel ett, dekrypteringen med nyckel två och ytterligare en kryptering med nyckel ett. Den sista metoden baseras på en enda nyckel. Syftet med denna metod är att få kompatibilitet med vanlig DES-kryptering. Nyckel ett krypterar, följt av att samma nyckel används vid dekryptering. Resultatet här är det ursprungliga meddelandet. Därefter sker kryptering en gång till med samma nyckel. Detta meddelande kan alltså dekrypteras med DES-algoritmen.

AES [7] är en relativt ny krypteringsalgoritm som utvecklades för att ett mer robust alternativ till DES behövdes. Ett av kraven för algoritmen var att den skulle vara säker i 20-30 år framåt, och de som lyckades bäst med detta var två belgiska kryptografer vars algoritm heter Rijndael. Algoritmen krypterar i 128 bitar stora block och stöder nyckelstorlekar på 128, 192 och 256 bitar.

3.3.3.5 Adressöversättning och IPsec

Ett känt problem som uppstår med IPsec är när *Network Adress Translation* (NAT) används, vilket innebär att du har lokala nätverksadresser inom ditt egna privata nät som inte kan adresseras direkt utifrån. Cisco har dock en färdig lösning till detta problem och den enda åtgärd som behöver göras är att aktivera IPsec över UDP. Vi har genomfört tester på detta och det fungerade utan anmärkning.

3.4 Behörighetskontroll

Det finns ett antal sätt att skapa behörighetskontroll via en ACL (se 3.2.4 Behörighetskontroll). Vid en implementation av ett VPN vore det en bra lösning att låta servern utnyttja intranätets befintliga metod för att ge behörighet åt fjärranvändare. Ofta har VPN-servern en inbyggd användardatabas, men detta innebär i mångt och mycket dubbelt arbete för administratören, eftersom många användare måste anges två gånger. Vidare har ofta VPN-servern inbyggt stöd för att kunna använda dessa tjänster som intranätet använder sig av,

och gränssnittet mellan dessa kallas *Remote Authentication Dial-In User Service* (RADIUS). Detta är ett sätt att erhålla autentiseringstjänster för många olika plattformar och klienter.

Som ett exempel kan nämnas att Novells senaste RADIUS-serverlösning heter *Border Manager Authentication Services* (BMAS), och denna kan användas av Ciscos koncentrator.

3.5 Prestandamätning

När kryptering används så sänks överföringshastigheten beroende på vilken algoritm och hur stor nyckel som används. För att få en uppfattning av hur stor påverkan krypteringen har utförde vi ett antal tester för några olika konfigurationer, där endast paketkrypteringen skiljer dem åt. Mätningarna genomfördes genom att överföra filer på cirka 10, 25 och 50 megabyte och tidsåtgången mättes med en digitalklocka. För varje testöverföring utfördes två mätningar, en från klient till det interna nätverket och en åt motsatt håll. I Tabell 3.1 nedan redovisas medelvärden av tiderna som erhölls med precision på 0,5 sekunder.

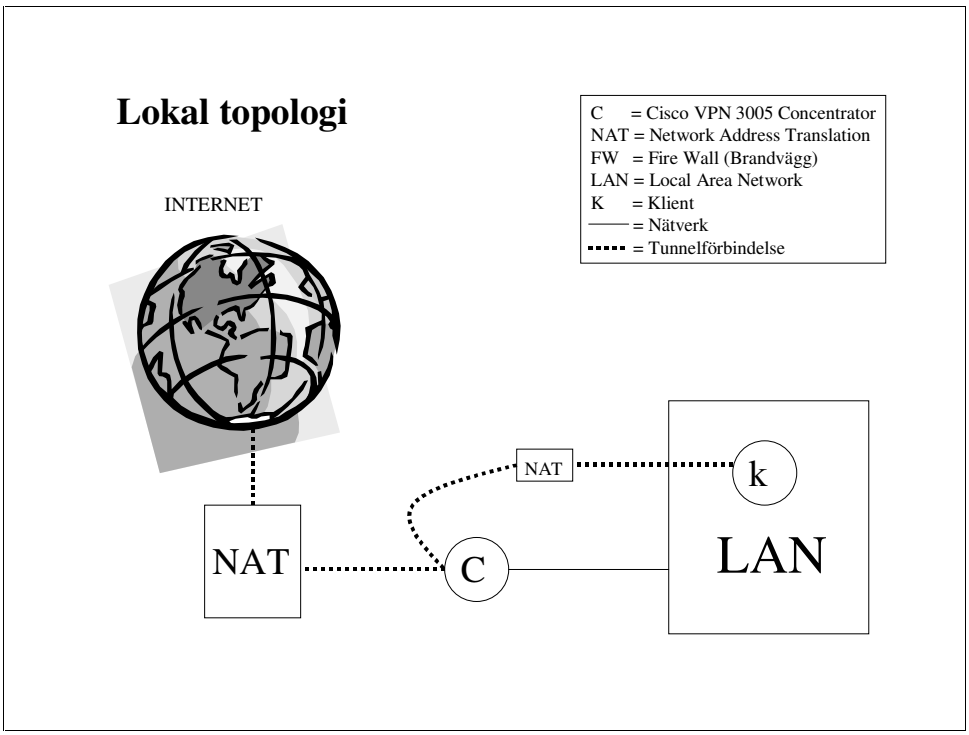
Tabell 3.1: Tidsåtgång beroende på filstorlek och algoritm

Algoritm	Tid för 10 MB (s)	Tid för 25 MB (s)	Tid för 50 MB (s)
3DES	25,0	63,0	125,5
DES	15,0	38,0	76,5
INGEN	3,5	8,5	16,0

Av resultaten framgår att det blir en markant prestandaförlust när kryptering används, dock skiljer det inte lika mycket mellan de båda krypteringsalgoritmerna. Det ser också ut som att tidsåtgången ökar linjärt med filstorleken vilket verkar vara logiskt. Den genomsnittliga överföringshastigheten utan kryptering hamnade på 24,1 Mbit/sekund. När kryptering aktiverades sjönk hastigheten till 5,3 Mbit/sekund för DES respektive 3,2 Mbit/sekund för 3DES. Det är alltså väsentligt att ta med detta i beräkningarna vid utvärdering av vilken algoritm som ska användas.

4 Konfigurationsexempel

Den hårdvara vi hade tillgång till i vår testmiljö på kommunen var en Cisco 3005 koncentrator, en Cisco 1600 router, en dator för konfigurering av koncentratorn och en dator för att koppla upp sig mot kommunens interna nät via koncentratorn. Konfigureringen av koncentratorn skedde via ett webbaserat gränssnitt som heter ”VPN 3000 Concentrator Series Manager”. VPN-klienten som vi använde för uppkoppling mot koncentratorn var Cisco Systems VPN Client Version 3.6.3. Vi provade med dessa hjälpmedel att koppla en klientdator direkt mot koncentratorn samt att låta klienten vara kopplad till koncentratorn via en router som använde sig av NAT (se 3.3.3.5 Adressöversättning och IPsec). De tester som gjordes för att ta reda på om datorerna kunde få kontakt med varandra efter uppkopplingsfasen bestod av filöverföringar mellan dem.



Figur 4.1: Skiss över testmiljö

4.1 Konfigurering av koncentrator och klient

För att komma igång med ett fungerande VPN ligger det allra största arbetet på koncentratorsidan. I koncentratorn anges hur VPN-förbindelserna ska upprättas. För att en användare ska kunna logga in måste den tillhöra en grupp. Det finns två olika slags grupper,

en basgrupp som fungerar som en standardgrupp och egenskapade grupper. De egenskapade grupperna kan ärva valfria egenskaper från basgruppen. Varje enskild användare ärver i sin tur egenskaper efter den grupp den tillhör. De inställningar som går att göra för den enskilda användaren har dock högre prioritet än gruppinställningarna.

Exempel på regler för grupperna är bland annat vilka protokoll som är tillåtna, hur användare autentiseras samt hur länge de får vara uppkopplade. Det finns två möjligheter att bestämma hur autentiseringen ska ske, den ena innebär att det sker med certifikat och den andra är en ”shared secret”-lösning. Att dessa inställningar sker i koncentratorn medför att den som vill koppla upp sig mot kommunens nät via koncentratorn inte kan påverka hur uppkopplingen skapas. Det är således en ganska enkel procedur att skapa en VPN-uppkoppling så länge alla parametrar är korrekt inställda hos kommunen, framförallt om certifikat inte används.

Om kommunen skulle vilja använda sig av certifikat finns det inte något behov att anlita en extern *Certificate Authority* (CA) i detta fall, då det är kommunens egna anställda som ska ha tillgång till nätet (se 2.3 Autentisering). Det är då en bättre lösning att installera en server lokalt med CA-mjukvara och låta den sköta certifikathantering. Detta innebär naturligtvis mer administrativt arbete, men bättre kontroll erhålles och abonnemang på en sådan tjänst behövs ej.

4.2 Konfiguration 1: Shared secret

Det enklaste sättet att skapa ett användarkonto är att använda ”shared secret”-lösningen. Ett användarnamn med lösenord måste skapas och detta kopplas till en grupp i koncentratorn. Motsvarande uppgifter anges i klientmjukvaran. Det är upp till administratören att skapa och delge användarna dessa uppgifter.

Denna konfiguration medför en av koncentratorns lägsta säkerhetsnivåer, då autentiseringen sker via ett överenskommet lösenord (se 2.1.1 Symmetrisk kryptering). Fördelen med denna konfiguration är att overhead-datan blir mindre vid autentiseringen, men framförallt är detta den lösning som medför minst arbete för både administratör och användare.

4.2.1 Inställningar i koncentratorn

4.2.1.1 Skapa grupp

Här följer nu ett exempel på hur en grupp kan skapas med en användare för inloggning i Karlstad kommuns intranät, vilket bilderna 4.2 och 4.3 visar:

Figur 4.2: Skapa en grupp

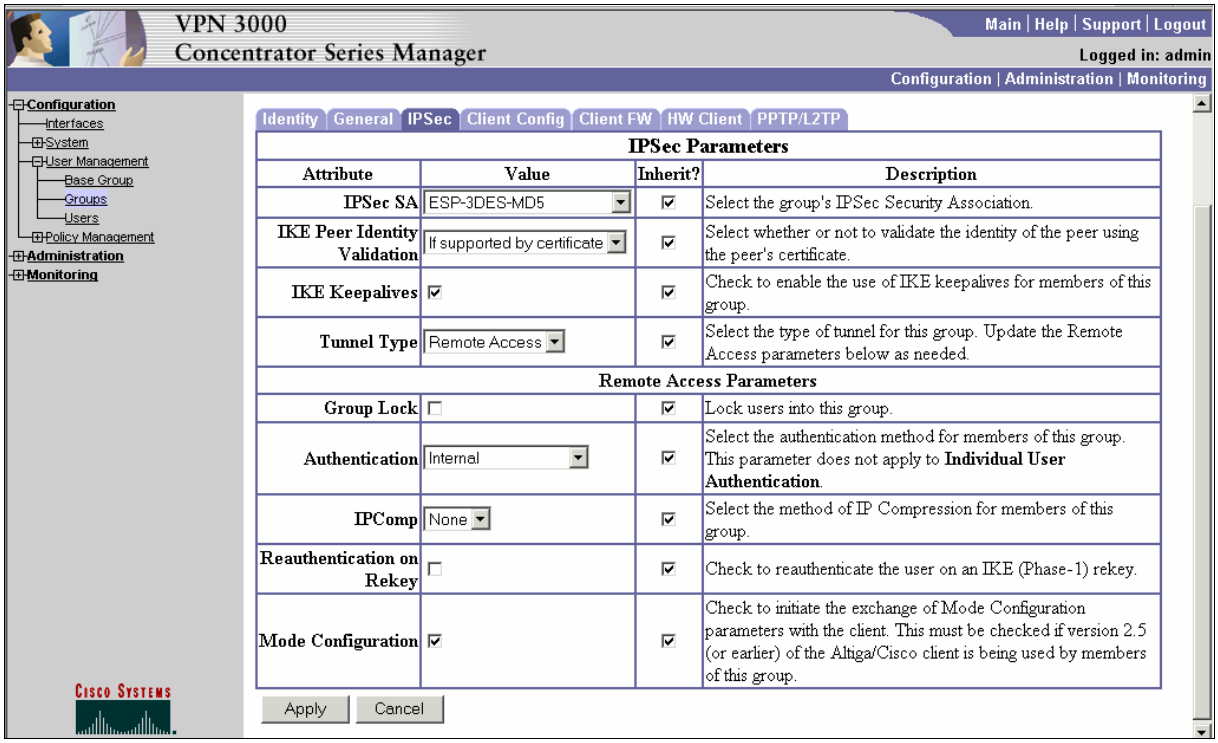
Administratören loggar in på koncentratorn via det webbaserade gränssnittet och går via trädet till vänster i bilden till **groups** (Configuration→User Management→Groups). Här väljs **Add Group**, och en ny skärmbild kommer fram. Denna bild har sju flikar med inställningar.

Efter varje inställningsmöjlighet följer en ruta under rubriken **Inherit**. Om denna ruta är bockad ärvs inställningen från basgruppen.

Den första heter **Identity**, och det är här namn och lösenord anges, samt vilken typ av autentisering, som ska gälla för gruppen. Vi valde att kalla gruppen *Anställda* och lösenordet *cisco123*. Autentiseringsfältet ger två val, **Internal** eller **External**. Om **Internal** väljs används koncentratorns interna databas, medan **External** ger möjlighet att använda tjänster som RADIUS. Då RADIUS ej är konfigurerat på intranätet är **Internal** det givna valet.

Flik nummer två kallas **General**. Här ställs allmänna parametrar in så som vilka tider inloggning är tillåten, maximalt antal samtidiga inloggningar, vilka tunnelprotokoll som tillåts med mera. Den enda ändringen vi gjorde från grundinställningen var att endast tillåta IPsec som tunnelprotokoll i **Tunneling Protocol**-fältet.

Under **IPSec** görs de inställningar som gäller specifikt för IPsec. Den översta inställningen anger vilken typ av **IPSec SA** som ska användas. Det finns ett antal förinställda att välja bland, men nya kan skapas med de inställningar som önskas (se 4.3.2.2 Skapa IPsec SA). SA:n anger vilken inkapslingsmetod, krypterings- och hashalgoritm som ska användas. I detta fall valde vi **ESP-3DES-MD5.** I denna flik anges även vilken typ av tunnel som ska användas, då det finns två olika att välja mellan, **Remote Access** och **LAN-to-LAN**. Den förstnämnda av dessa används om det är användare som ska koppla upp sig mot koncentratorn, då den andra endast gäller om två intranät ska kopplas samman. Därför valde vi i detta fall **Remote Access**. Vidare bestäms vilken autentiseringsmetod som ska användas för IPsec-tunneln. Exempel på valbara metoder är **None**, **RADIUS** och **Internal**. Alternativet **None** ger ingen autentisering, **RADIUS** utnyttjar en extern databas och **Internal** använder koncentaratorns inbyggda databas. Även här valde vi **Internal**.



Figur 4.3: IPsec-konfiguration för grupp

Under **Client Config** begränsas valmöjligheterna för klienten. Där finns tre underavdelningar, Cisco Client Parameters, Microsoft Client Parameters och Common Client Parameters. För Cisco-klienter kan välkomstmeddelande anges vid inloggning, vilket skrivs i **Banner**-fältet. Om **Allow Password Storage on Client** aktiveras ges användaren möjlighet att spara lösenorden lokalt på sin dator, vilket kan äventyra säkerheten. Även IPsec över UDP kan tillåtas, vilket används om klienten befinner sig bakom en NAT-router. Detta görs genom

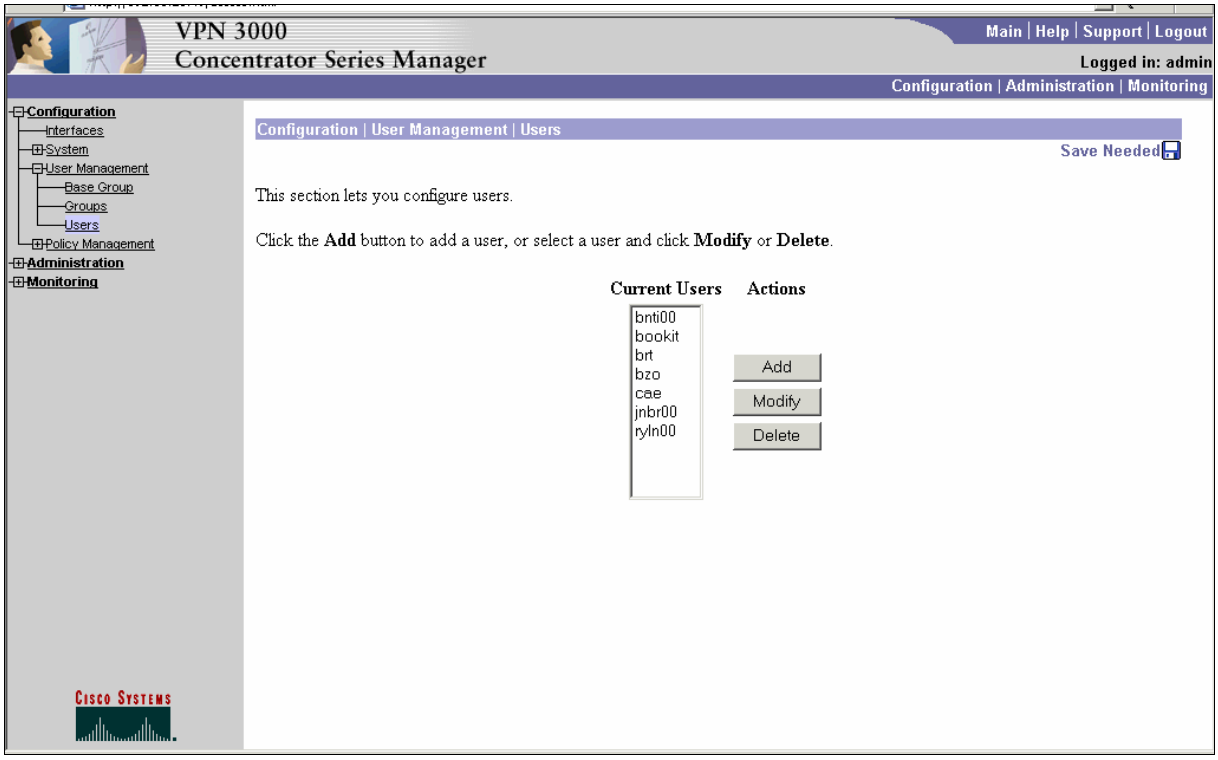
att bocka för **IPSec Over UDP**. I Common Client Parameters-fältet är den viktigaste inställningen att ta ställning till om split tunneling ska tillåtas. Vi valde här att inte tillåta detta vilket innebär att all data tunnlas och säkerheten höjs. Det fält som styr detta kallas **Split Tunneling Policy**, och den inställning vi valde var **Tunnel everything**.

Därefter kommer fliken **Client FW**. Här ställs klientens brandväggskonfiguration in, det vill säga om det krävs att klienten har en brandvägg eller ej. Några av de förkonfigurerade brandväggarna är **Cisco Integrated Client Firewall**, **Network ICE BlackICE Defender** samt **Zone Labs ZoneAlarm**. Om klientens brandvägg inte finns listad kan i stället **Custom Firewall** väljas och egna inställningar anges. Cisco Systems VPN Client 3.6.3(A) har en inbyggd brandvägg, men denna används endast vid split tunneling. Därför angav vi **No Firewall** för denna anslutning.

Den näst sista fliken heter **HW Client** och gäller endast i de fall dedikerad hårdvaruklient används. Detta var inte fallet för oss, därför lät vi standardvärdena stå kvar.

Till sist finns fliken **PPTP/L2TP**. På denna sida görs inställningar för de två återstående tunnelprotokoll som koncentratorn kan använda. Vi valde i detta fall att endast använda oss av IPsec, och därmed är denna flik inte intressant för denna konfiguration.

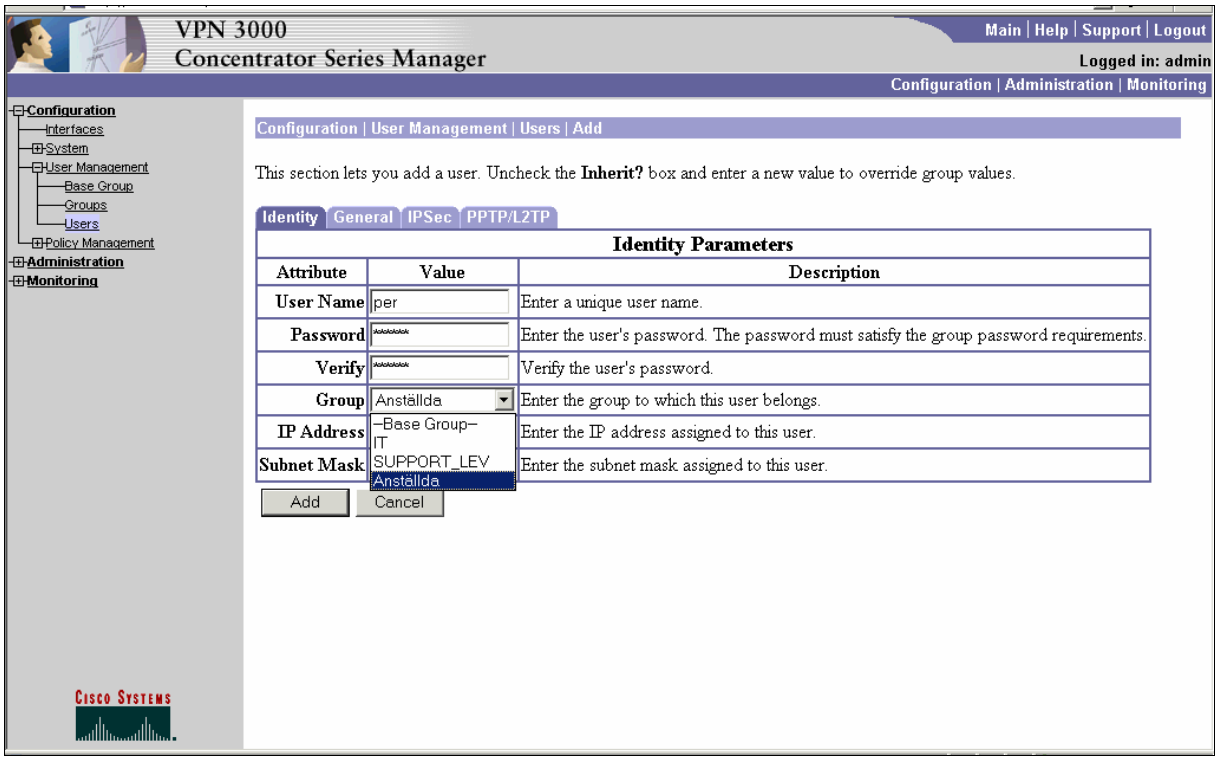
4.2.1.2 Skapa användare



Figur 4.4: Skapa användare

Efter att gruppen skapats är nästa steg att skapa användare, vilket sker under **Users**, se Figur 4.4. Knappen **Add** leder till en ruta för konfigurering av ny användare. Precis som när en grupp skapas, som ärver av basgruppen, ärver även användaren inställningarna av sin egen grupp.

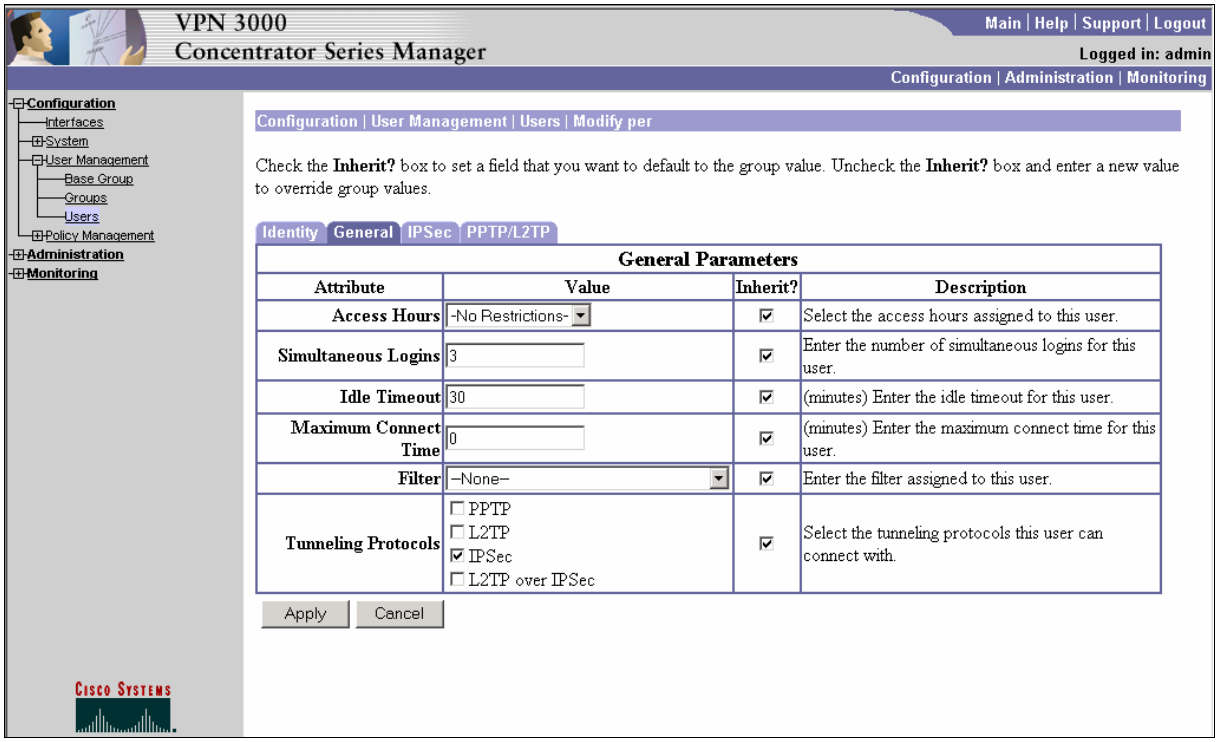
Den första fliken benämns, precis som för grupper, **Identity**, och kan beskådas i Figur 4.5. Användarnamn, lösenord samt grupptillhörighet anges här. Även IP-adress och nätmask kan anges, som endast ska gälla för denna användare. Då åsidosätts den på förhand konfigurerade adress-poolen.



Figur 4.5: Ange användarnamn och grupptillhörighet för användare

Vi valde i detta fall att kalla användaren för *per* och lösenordet blev återigen *cisco123*. *per* kopplades sedan till gruppen *Anställda*.

Under **General**-fliken återfinns några av de inställningsmöjligheter som finns under motsvarande flik för grupper. I Figur 4.6 har användaren ärvt alla egenskaper av tillhörande grupp.

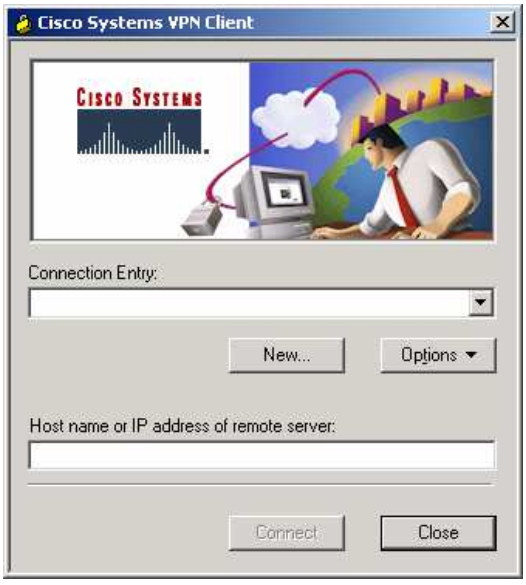


Figur 4.6: Inställningar under General-fliken för användare

På liknande sätt ärvs egenskaperna under **IPSec-** och **PPTP/L2TP**-flikarna.

4.2.2 Inställningar i klienten

När Ciscos klientmjukvara startas möts användaren av följande fönster:



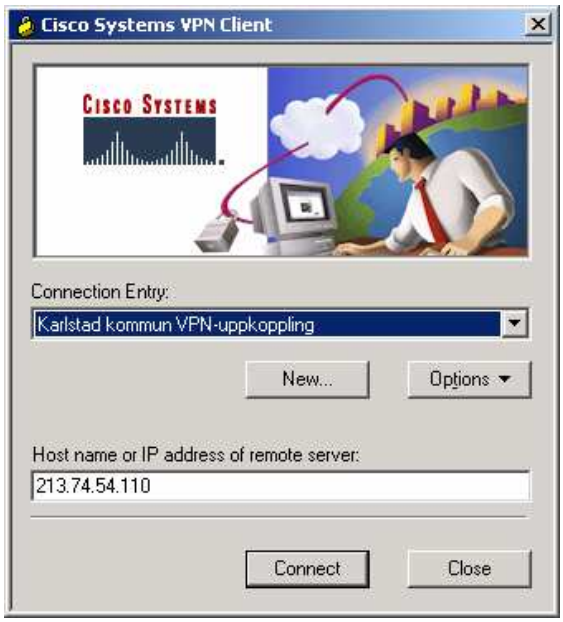
Figur 4.7: Cisco VPN Client, första uppstarten

Det finns ingen förinställd uppkoppling. En ny sådan skapas med knappen **New**. Det första som ska anges i den ruta som kommer upp är namnet på uppkopplingen samt en beskrivning av den om så önskas. Vi valde i detta fall att kalla uppkopplingen för ”Karlstad kommun

VPN-uppkoppling” och angav ingen beskrivning. Ett tryck på knappen **Nästa** för användaren till en ny ruta där adressen till VPN-servern anges. I rutan efter denna anges grupptillhörighet för användaren, se Figur 4.8.

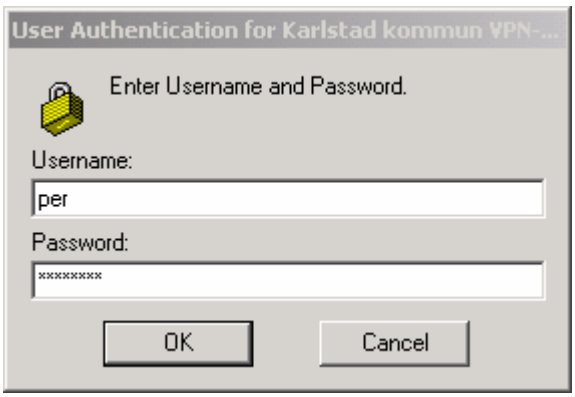
Figur 4.8: Cisco VPN Client, val av autentiseringsmetod,” shared secret”

Eftersom vi i koncentratorn uppgav att användaren *per* hör till gruppen *Anställda*, skrivs detta in i fältet **Name**. Lösenordet är det på förhand överenskomna *cisco123*. Ett tryck på knappen **Nästa** leder till en ruta där konfigurationen endera godkänns eller förkastas. Ett tryck på knappen **Slutför** gör att guiden försvinner och användaren återkommer till startfönstret. Den enda skillnaden mot förut är att nu finns det en uppkoppling att välja under **Connection Entry**, se Figur 4.9. Namnet är det som angavs i guiden, ”Karlstad kommun VPN-uppkoppling”.



Figur 4.9: Cisco VPN Client, uppkoppling färdigkonfigurerad

Uppkoppling sker vid ett tryck på knappen **Connect**. Då kommer en ny ruta fram där det individuella användarnamnet och lösenordet anges:



Figur 4.10: Cisco VPN Client, ange anvädarnamn och lösenord

Användarnamnet är *per* och lösenordet *cisco123*. Knappen **OK** leder till att klienten initierar uppkopplingen mot koncentratorn. Efter att alla verifieringar är klara är nu klienten uppkopplad mot företagets intranät och kan använda detta.

4.3 Konfiguration 2: Certifikat

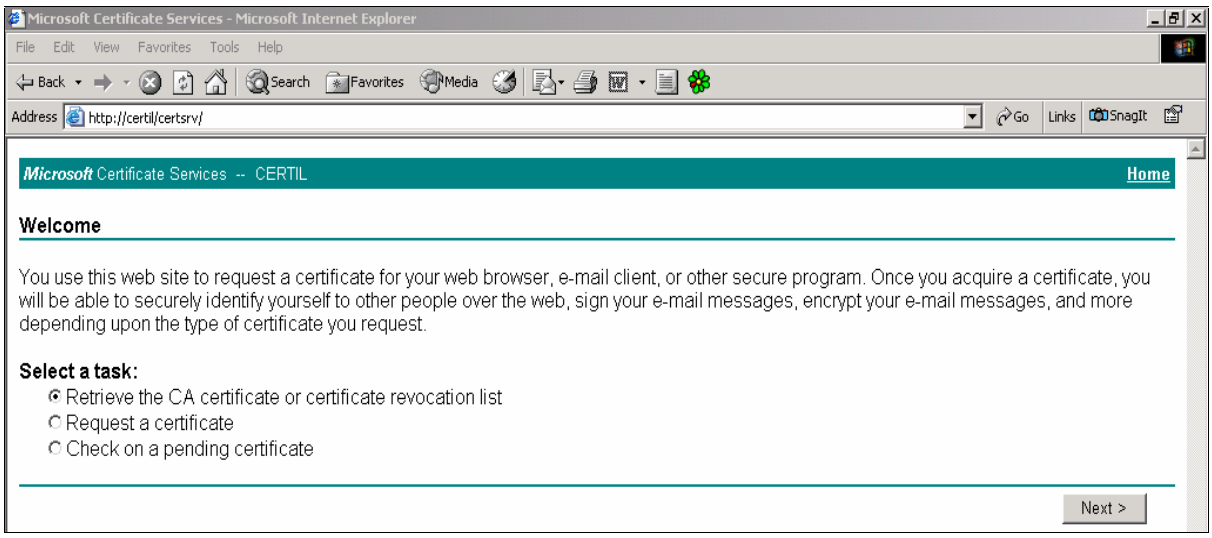
För att få en säkrare VPN-förbindelse används certifikat i denna konfiguration. Detta medför betydligt mer arbete och högre kompetenskrav om kommunen väljer att vara sin egen CA (se 2.3 Autentisering). Det andra alternativet är att köpa tjänsten från en utomstående CA. I följande konfiguration används det förstnämnda alternativet. För att denna lösning ska kunna

genomföras krävs det att både klientdator och koncenterator installerar ett CA-certifikat från samma CA, ett så kallat rotcertifikat. Vidare krävs det även ett unikt certifikat per maskin för att bekräfta dess identitet. Distributionen av certifikaten kan ske på flera olika sätt och hur detta ska ske är upp till den säkerhetsansvariga att ta ställning till. Vi kommer för detta exempel att skicka certifikaten via e-post för enkelhetens skull men det är inget att rekommendera ur säkerhetssynpunkt. Den kanske säkraste metoden torde vara distribution via flyttbart medium såsom diskett eller liknande.

4.3.1 Installation av certifikat

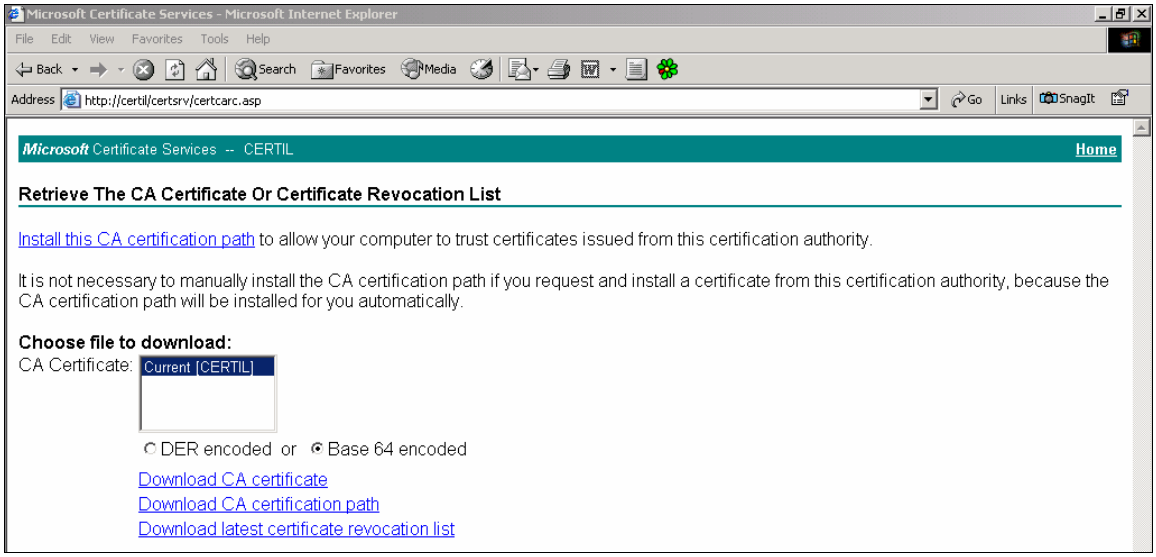
4.3.1.1 Erhållande av rotcertifikat från certifikatservern

Certifikatservern har ett webbgränssnitt som är kopplat till certifikattjänsten. Denna kan användas för att hämta rotcertifikat (se 2.3 Autentisering), begära identitetscertifikat samt att hämta detta certifikat när det skapats. Den första bilden användaren möter ser ut enligt följande:



Figur 4.11: Certifikatserver, skapa förfrågan

För att hämta rotcertifikat görs en förfrågan till certifikatservern genom att välja **Retrieve the CA certificate or certificate revokation list** enligt Figur 4.11 och trycka på knappen **Next**. Efter att detta har utförts möts användaren av ett nytt fönster där korrekt rotcertifikat väljs och därefter vilken information som önskas vilket framgår av Figur 4.12.

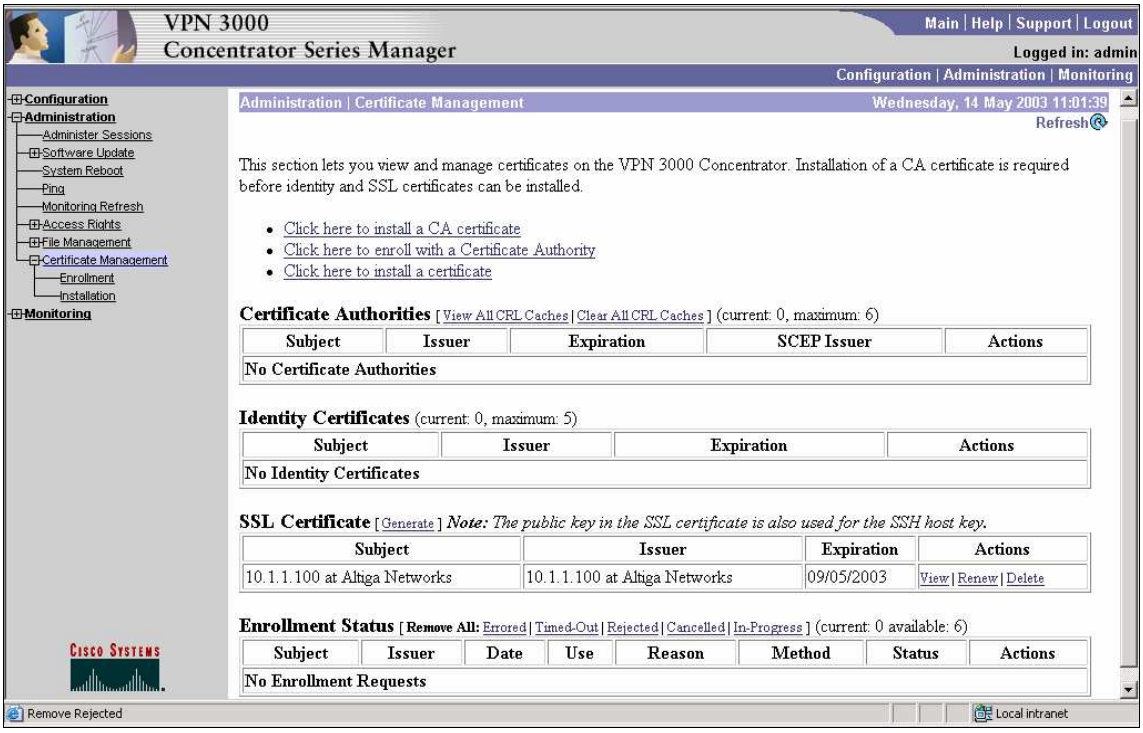


Figur 4.12: Certifikatserver, hämta rotcertifikat

De tre alternativen att välja på är: **Download CA certificate**, **Download CA certification path** och **Download latest certificate revokation list**. Därutöver finns två alternativ för att format på certifikatfilen, ”Base 64” och ”DER”. Valet ”Base 64” innebär att filen kodas enligt ASCII-standard och går att öppna i en textredigerare. ”DER” kodar filen binärt, vilket gör att informationen inte går att läsas i en textredigerare. Anledningen till att den förstnämnda existerar är att en metod för installation av certifikat baseras på att texten klipps ut från filen och skickas till önskad mottagare.

4.3.1.2 Installation av rotcertifikat i koncentratorn

Rotcertifikatfilen skickas till koncentratorns administrationsdator och administratören loggar in på VPN-servern. Nästa steg är att installera certifikatet vilket sker via länken **Click here to install a CA certificate** (Administration→Certificate Management) enligt Figur 4.13.



Figur 4.13: Koncentrator, installation av certifikat

Länken leder till ett nytt fönster där metod för installation av certifikatet väljs. Vilket av alternativen som är aktuellt beror på önskad metod och vilket format certifikatfilen har.

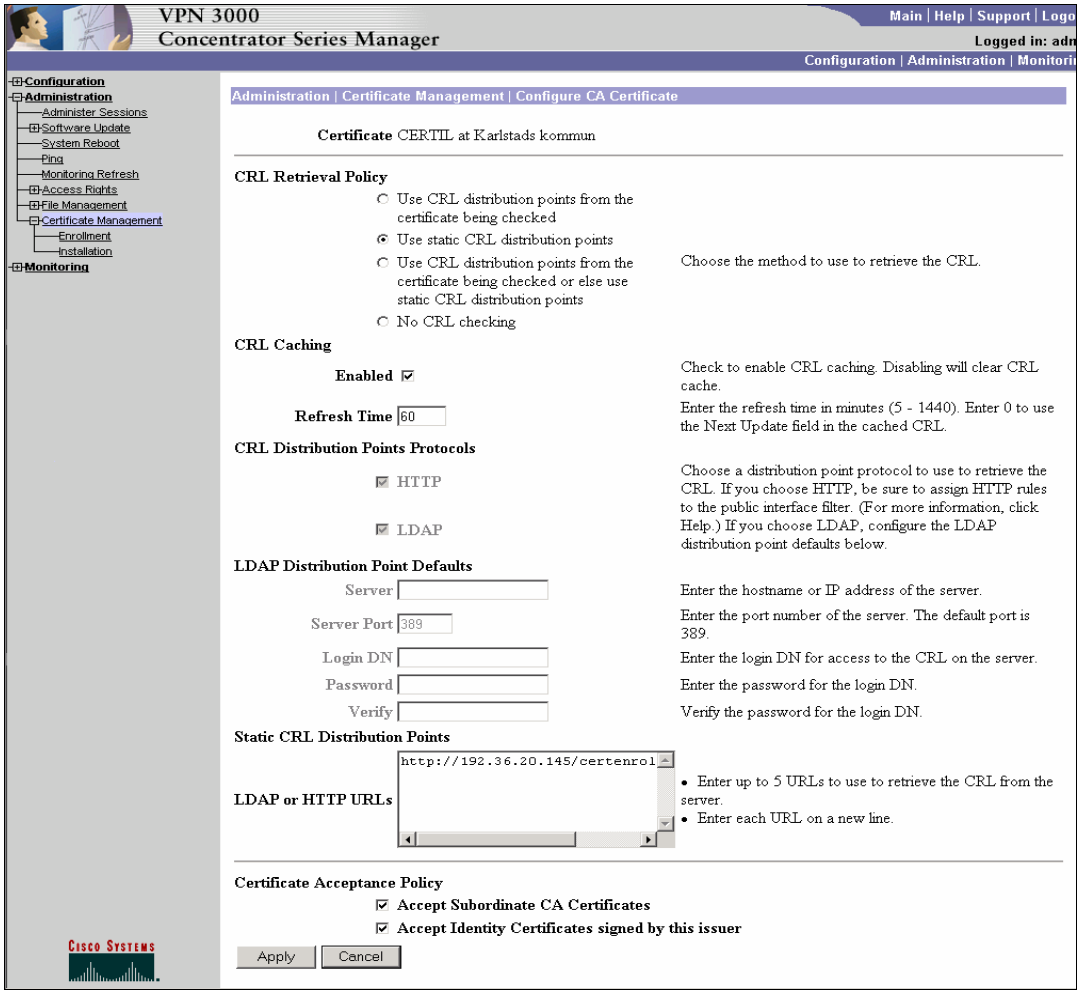
Om filen är en ”DER”-fil så är det bara valet **Upload File from Workstation** som är aktuellt. Om certifikatet är en ”Base 64”-fil fungerar både detta alternativ och **Cut & Paste Text**. Det tredje alternativet, **SCEP (Simple Certificate Enrollment Protocol),** kan endast användas om certifikatservern stödjer detta.

Väljs **Cut & Paste** visas en tom ruta med beskrivningen ”Certificate Text”. Öppna textfilen med certifikatet i en textredigerare och kopiera över texten till denna ruta, vilket har skett i Figur 4.14.

Figur 4.14: Koncentratorn, klistra in rotcertifikattext

Tryck sedan på **install**. Om istället **Upload File from Workstation** väljs får administratören ange sökväg till certifikatfilen, som därefter installeras. Nu har rotcertifikatet för CA:n installerats och koncentratorn därmed litar på den.

Certifikathanteringen innehåller funktioner för att undersöka vilka certifikat som blivit återkallade. Dessa inställningar görs via länken **Configure** för rotcertifikatet (Administration→Certificate Management) och återspeglas i Figur 4.15. Den första inställningen gäller vart koncentratorn ska leta efter *Certificate Revokation List* (CRL), eller om sådan ens ska användas. Standardalternativet är **No CRL checking**, men det är god praxis att använda sig av denna funktion när så är möjligt. Då vi i denna konfiguration endast har en certifikatserver används **Use static CRL distribution points**. Därefter ges möjligheten att lagra denna lista lokalt på koncentratorn samt ange hur ofta den ska uppdateras. Detta val sker under ”CRL Caching”, och vi valde att använda oss av detta. Under detta val anges hur ofta, i minuter, listan ska uppdateras. Här kallas detta ”Refresh Time”, och vi gav denna inställning värdet 60. Till sist måste sökvägen till listan anges, vilket sker längst ner under ”LDAP or HTTP URLs”.



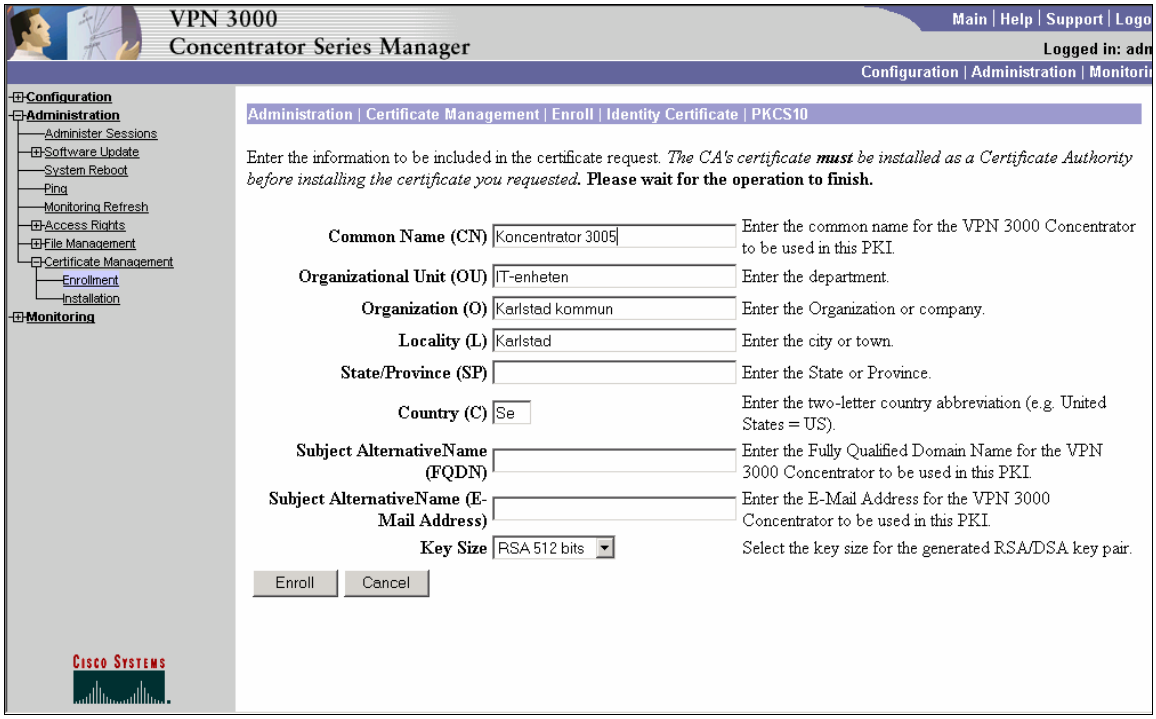
Figur 4.15: Koncentrator, ange CRL distribution points

Vi använder oss av http i detta fall, och för att få reda på sökvägen följs länken **View** för rotcertifikatet (Administration→Certificate Management). Denna länk finns till vänster om **Configure**, vilken används här ovan. Längst ner finns sökvägen, som här kallas just ”CRL Distribution Point”.

4.3.1.3 Förfrågan av identitetscertifikat för koncentratorn

Nästa steg i processen är att installera ett identitetscertifikat som är unikt. Detta certifikat används sedan för att klienter ska kunna identifiera koncentratorn. Administratören väljer **Click here to enroll with a Certificate Authority** (Administration→Certificate Management) och kommer till en ny ruta där det finns två val, ”Identity certificate” och ”SSL certificate”. I detta fall används **Identity certificate**. Därefter finns endast ett val att göra, **Enroll via PKCS10 Request (Manual)**. Om den certifikatserver, vars rotcertifikat installerades under rubrik 4.3.1.2, har stöd för *Simple Certifikate Enrollment Protocol* (SCEP) kan en automatisk procedur väljas. Då detta stöd inte finns är det inte aktuellt i detta fall. Efter

att ha valt den manuella metoden möts administratören av en ruta där information om koncentratorn anges som i Figur 4.16.



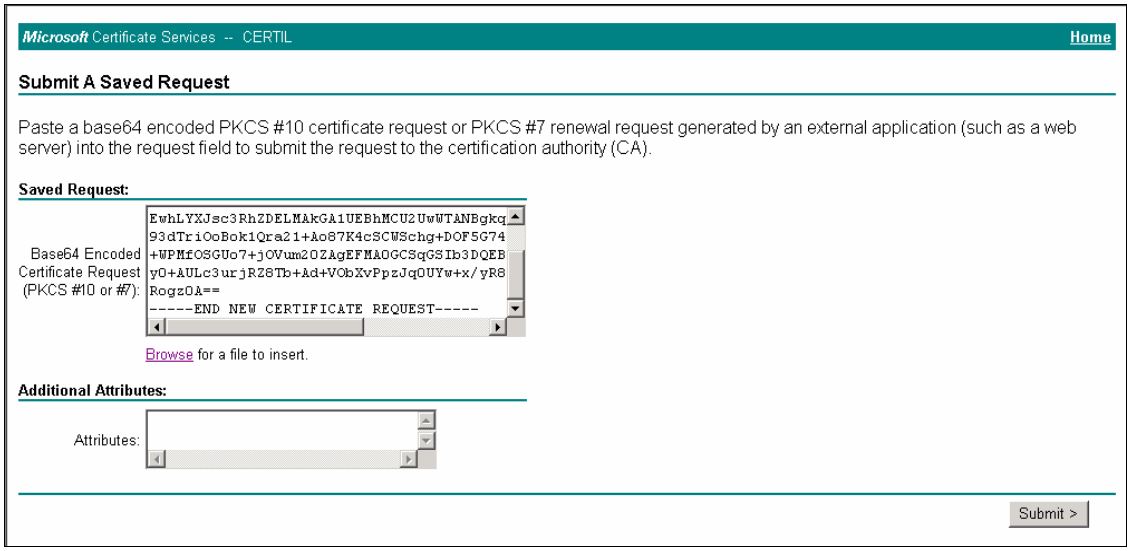
Figur 4.16: Koncentrator, ange certifikatinformation

Som ”Common Name” angav vi *Koncentrator 3005*, ”Organizational Unit” *IT-enheten*, ”Organization” *Karlstad kommun*, ”Locality” *Karlstad* och ”Country” *Se*. Den enda information som *måste* anges är ”Common Name”. De andra är valfria, men bör anges då det underlättar vid administration. Därefter måste algoritm samt nyckelstorlek väljas. Om inget aktivt val görs kommer ”RSA 512 bits” att användas, men DSA-algoritmen kan också väljas. Nycklarna finns i storlekar mellan 512 och 2048 bitar. Vi valde här att använda standardalternativet ”RSA 512 bits”. Knappen **Enroll** leder till att en ny ruta kommer fram där den genererade certifikatförfrågan visas. Denna förfrågan visas i klartext och ska kopieras in i en textfil för vidarebefordran till certifikatservern. Vi valde här att skicka förfrågan via epost.

4.3.1.4 Erhållande av identitetscertifikat från certifikatservern

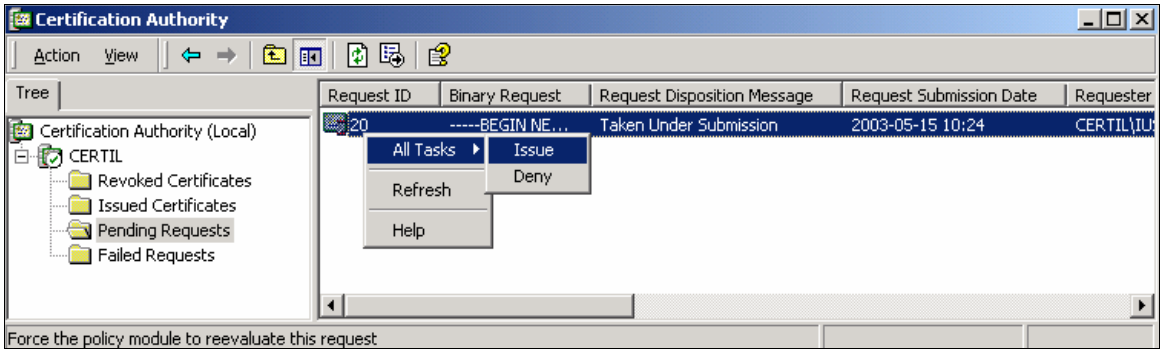
När en förfrågan kommer till certifikatadministratören loggar den in på certifikatservern och väljer **Request a certificate**. I nästa fönster som kommer upp väljs **Advanced request**. Därefter väljs “Submit a certificate request using a base64 encoded PKCS#10 file or a renewal request using a base64 encoded PKCS#7 file”. Rutan som nu kommer fram innehåller två fält, “Saved Requests” och ”Additional Attributes”. I detta fall används endast

det första fältet. Det är här som förfrågan klistras in, endera från en textredigerare eller så anges sökväg till filen, se Figur 4.17. Det förstnämnda är en bra metod om webbläsarena säkerhetsinställningar inte tillåter att filer läses från hårddisken. Då detta gäller i vårt fall använder vi oss av denna ”klipp och klistra”-metod.



Figur 4.17: Certifikatserver, klistra in en förfrågan

När texten klistrats in kommer en ruta fram där det står att certifikatet mottagits. Därefter måste administratören använda sig av ett verktyg som heter ”Certification Authority”. (Start→programs→Administrative Tools) vilket ser ut som i Figur 4.18. Här finns ett antal undermenyer, där den i detta fall intressanta är ”Pending Requests”.



Figur 4.18: Certification Authority, godkänn certifikat

Här finns nu förfrågan på certifikatet, och det kan godkännas eller nekas. I detta fall godkänns certifikatet. Det hamnar då under ”Issued Certificates” och certifikatet kan hämtas från webbgränssnittet. **Check on a pending certificate** bockas för, och nästa ruta visar en lista över sparade certifikatförfrågan. Det aktuella certifikatet väljs, och leder till ytterligare en ruta där information om status anges. Om certifikatet har godkänts är det nu möjligt att ladda ner det. Vi valde att ladda ner den binärkodade filen (DER), då denna metod är enklast. Det

andra valet är ”Base 64”, det vill säga ASCII-kodad fil, och är främst avsett för ”klipp och klistra”-metoden.

Certifikatet skickas därefter till avsedd mottagare.

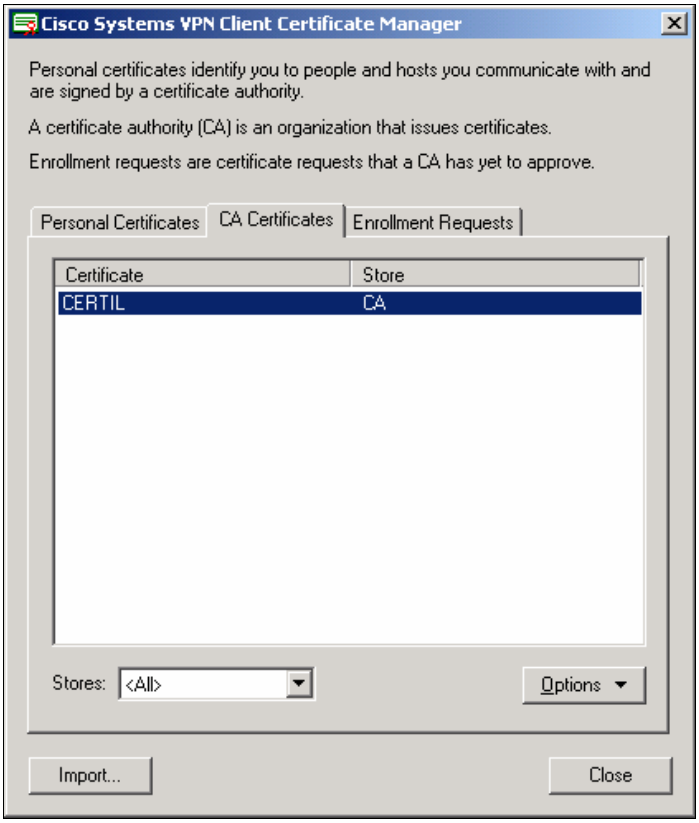
4.3.1.5 Installation av identitetscertifikat på koncentratorn

När certifikatet har utfärdats måste det installeras i koncentratorn. Detta sker via länken **Install** (Configuration→Certificate Management). Länken finns längst ner till höger, under ”Enrollment”. Här kan det ligga flera certifikatförfrågningar, om certifikatservern ska ha flera identitetscertifikat (Cisco VPN Concentrator 3005 kan ha upp till fem sådana installerade samtidigt). Länken **Install** finns då under ”Actions” vid det korrekta certifikatet. Denna länk leder till en ruta med två länkar, ”Cut & Paste Text” och ”Upload File from Workstation”. Eftersom vi sparade certifikatet i DER-formatet måste vi nu välja det senare alternativet. Den nya rutan som kommer fram innehåller ett fält för sökväg till filen. När denna angivits installeras certifikatet. Därefter återgår webbläsaren till huvudmenyn för certifikathanteringen. Nu återfinns certifikatet under ”Identity Certificates”.

När certifikatet installeras sparas en kopia av det i filsystemet på koncentratorn. Denna fil bör tas bort när certifikatet är klart att användas. Detta sker via länken **Delete** för avsedd fil (Administration→File Management).

4.3.1.6 Installation av rotcertifikat i VPN-klienten

Begäran av rotcertifikatet sker enligt 4.3.1.1. När rotcertifikatfilen erhållits öppnas klientens ”Certificate Manager”. För att installera ett rotcertifikat väljs **Import.** Ett nytt fönster öppnas och alternativen ”Microsoft certificate” och ”File” visas under ”Certificate source”-rutan. Under ”File” anges sökvägen till rotcertifikatet. Följande fönster tillåter klienten att binda ett lösenord till certifikatet som kommer att krävas varje gång certifikatet används. Detta är valfritt och inte aktuellt för ett rotcertifikat.



Figur 4.19: Cisco Certificate Manager, rotcertifikat installerat

Rotcertifikatet är nu installerat vilket indikeras under fliken ”CA certificates” som i Figur 4.19.

4.3.1.7 Förfrågan av identitetscertifikat för VPN-klienten

Även i detta fall används Ciscos ”Certificate Manager”. Under fliken ”Personal Certificates” väljs **New**. Fönstret som visas då ger möjligheten att lösenordsskydda certifikatet. Vi valde här att inte ange något lösenord. Nästa val är på vilket sätt förfrågan ska ske, via nätverk eller fil. I detta fall väljer vi filmetoden då certifikatservern är placerad i det privata nätverket och ej kan nås utifrån, vilket skulle krävas för nätverksmetoden. Vidare väljs namn på filen samt vart den skall sparas. Här väljs även vilket format filen ska sparas med.

När detta är gjort kommer ett fönster upp där personlig information av användaren anges. Som ”Common Name” angav vi *Per Johansson*, ”Organizational Unit” *IT-enheten*, ”Organization” *Karlstad kommun* och ”Country” *Se*. Den enda information som *måste* anges är även i detta fall ”Common Name”. När detta angivits kommer en ruta för godkännande av certifikatförfrågan.

Filen med förfrågan ligger nu på angiven plats och kan vidarebefordras till certifikatadministratören. Certifikatet skapas på det sätt som anges under rubriken 4.3.1.4.

4.3.1.8 Installation av identitetscertifikat på VPN-klienten

Efter att klientcertifikatet erhållits ska det installeras. Även detta sker i ”Certificate Manager”. Proceduren är likadan som för rotcertifikatet (se 4.3.1.6 Installation av rotcertifikat i VPN-klienten). Skillnaden ligger i att om ett lösenord angavs vid skapandet av certifikatförfrågan måste detta lösenord anges även här. Därefter är det personliga identitetscertifikatet installerat och redo att användas.

Precis som för koncentratorn är det en bra idé att ta bort certifikatfilen från hårddisk eller liknande, då den kan utgöra en säkerhetsrisk om någon obehörig kommer över den.

4.3.2 Konfigurering av koncentrator

För att få en förbindelse upprättad med användande av certifikat krävs det en egenkonstruerad IPsec SA då de förinställda endast fungerar med ”shared secret”-lösningen. Detta beror på att koncentratorns identitetcertifikat måste vara installerat då en av inställningarna som krävs är vilket certifikat som ska användas.

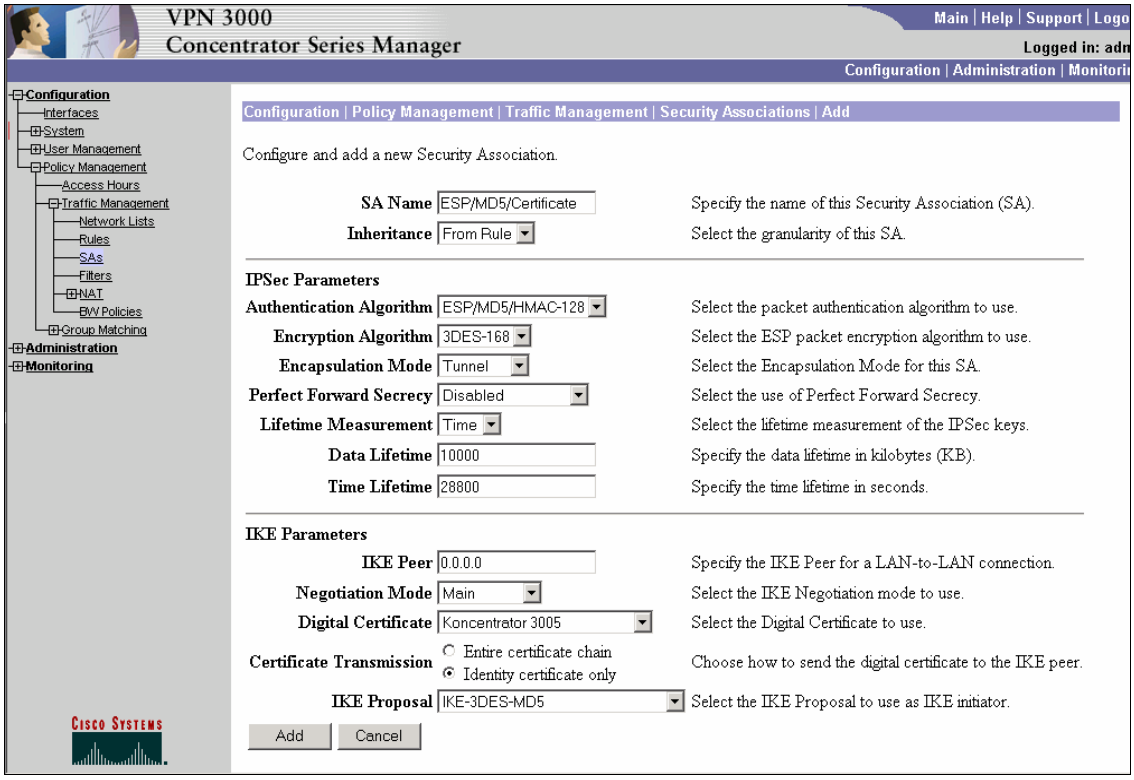
4.3.2.1 Aktivera avsett ”IKE proposal”

När certifikat används krävs det att det finns ett för ändamålet fungerande ”IKE proposal” under ”Active Proposals” (Configuration→System→Tunneling Protocols→IPsec→IKE Proposals). Med fungerande avses här att dess inställningar stödjer certifikat. Det finns förkonfigurerade ”IKE proposals” under ”Inactive Proposals” som inte tas hänsyn till förrän de är aktiverade.

I detta fall finns en förkonfigurerad ”IKE proposal” från grundinställningen som vi aktiverade genom att markera den och tycka på **Activate**. Denna heter ”CiscoVPNClient-3DES-MD5-RSA”.

4.3.2.2 Skapa IPsec SA

I föregående konfigurationsexempel (se 4.2 Konfiguration 1: Shared secret) anges en förkonfigurerad IPsec SA under rubriken 4.2.1.1. I detta fall finns ingen sådan, alltså måste en skapas. Detta sker genom att trycka på **Add** i SA-konfigureringsrutan (Configuration→Policy Management→Traffic Management→SAs) enligt Figur 4.20. Då ges möjlighet att namnge denna specifika SA samt ange alla inställningar för den.



Figur 4.20: Skapa egen IPsec SA

Koncentratorns standardalternativ passade vår konfiguration i stor utsträckning. Namnet blev ”ESP/MD5/Certificate”. Vi ändrade endast på inställningarna under ”IKE Parameters”. Som ”Digital Certificate” angav vi det certifikat vi redan installerat (se 4.3.1 Installation av certifikat), vilket i detta fall heter *Koncentrator 3005*. Vi valde även att endast skicka med identitetscertifikatet, då certifikatkedjan är onödig. Det sistnämnda beror på att certifikatservern är rotserver och inte har några underordnade certifikatservrar som delar ut certifikat. ”IKE Proposal” är inte intressant i denna konfiguration, utan används endast då koncentratorn ska initiera en uppkoppling mot en annan VPN-server.

När dessa inställningar är skapade är denna IPsec SA redo att användas.

4.3.2.3 Matcha användare mot grupper

I konfiguration ett kopplades användaren till en viss grupp genom att detta angavs när användaren skapades. Så fungerar det inte när certifikat används. Det finns då istället tre andra sätt att koppla en användare till en grupp, varav en eller fler måste vara aktiva. Dessa metoder, i prioritetsordning, är regler, avdelning och standardgrupp.

Om regler används måste sådana först skapas. De baseras på att det jämförs ett valfritt antal identitetsfält från ett certifikat och därefter kopplas användaren till den avsedda gruppen.

Till exempel kan regeln jämföra om namnet (CN) motsvarar visst namn och att denna person arbetar på en viss avdelning (OU) och genom detta avgöra vilken grupp klienten tillhör.

Att istället dela in användarna efter avdelning innebär att koncentratorn endast tittar på avdelningsfältet i certifikatet och kopplar användaren till den grupp som heter likadant.

Det sistnämnda alternativet är att samtliga certifikatanvändare kopplas till en och samma grupp vilken fritt kan väljas bland existerande grupper.

4.3.2.4 Skapa grupp

Även denna konfiguration, baserad på certifikat, använder sig av grupper och användarnamn. Tillvägagångssättet beskrivs i föregående konfiguration (se 4.2.1.1 Skapa grupp). Den enda skillnaden gäller för fliken **IPsec**, där ”IPSec SA” då istället ska vara den egenkonfigurerade ”ESP/MD5/Certificate”, och att viss möda läggs på ett genomtänkt gruppnamn. Det sistnämnda beror på att det kanske vanligaste sättet att koppla användare till grupper är via avdelningsfältet i certifikatet. Vi valde att kalla gruppen *IT-enheten*. Övriga inställningar är identiska med de i konfiguration ett. Dock kan det tilläggas att lösenordet för gruppen aldrig behöver anges vid uppkoppling, endast användarens personliga lösenord.

4.3.2.5 Skapa användare

Precis som för gruppen är skapandet av användare i mycket stor utsträckning likadan som i konfiguration ett. Vi valde att kalla användaren *Per Johansson*. Som personligt lösenordet valde vi *cisco123*. Detta lösenord måste anges vid uppkoppling, precis som tidigare. Vi kopplade användaren till gruppen *IT-enheten*.

4.3.3 Konfiguration av klient

Vad som behövs göras i klienten är att skapa en ny ”Connection entry” vilket inte skiljer sig mycket från motsvarande inställningar under 4.2.2. Den enda punkten som de skiljer åt är när fönstret för val av hur autentiseringen ska ske kommer upp. Här bockade vi för **Certificate** och valde certifikatet med titeln ”Per Johansson (Cisco)” som i Figur 4.21.

Figur 4.21: Cisco VPN Client, val av autentiseringsmetod, certifikat

Därefter är det bara att godkänna konfigurationen och precis som i konfiguration ett återstår endast att koppla upp VPN-förbindelsen.

5 Summering

I denna rapport har vi utrett bakomliggande begrepp för hur ett VPN fungerar och även skapat en guide för att konfigurera en fungerande uppkoppling med hjälp av Cisco-produkter.

Den första tiden använde vi till att leta information om VPN men insåg snart att det inte fanns lämplig litteratur att få tag på lokalt. Därför har vi fått det mesta av vår information via Internet. Nackdelen med detta är att informationen som vi fann sällan var på en lämplig nivå utan ofta var för kortfattad eller för närliggande kodimplementation. Vi sökte mestadels efter information som beskrev vad funktionerna utförde och inte hur. Sedan fortsatte vi arbetet med att sätta oss in i Ciscos VPN-lösning. Avslutningsvis skapade vi konfigurationsguiden. Vi kom fram till följande:

- VPN är ett mycket stort och komplext begrepp. De många protokollen och säkerhetsmekanismerna gör det svårt att få en enkel helhetsbild.
- En viktig faktor att ta hänsyn till är hur prestanda påverkas av säkerhetsnivån. Detta framgår av testet i kapitel 3.5. Ju högre säkerhet, desto långsammare uppkoppling.
- Distribution av certifikat är den svaga länken i certifikathanteringen då det är viktigt att ingen obehörig får tillgång till certifikaten. För att lösa detta kan distributionen exempelvis ske via flyttbart medium så som diskett, eller från en hemsida med lösenord.
- Ett krav innan ett VPN tas i bruk är att säkerhetskraven på förhand noga specificeras.
- De tester som vi genomfört har fungerat väl i testmiljön och om det inte uppstår andra komplikationer i ”verklig” miljö verkar VPN vara en mycket bra lösning på Karlstad kommuns problem.

Den naturliga fortsättningen på VPN-projektet borde vara att konfigurera en RADIUS-lösning samt hitta en lämplig rutin för certifikathanteringen. Därutöver bör möjligheten att kunna ställa krav på klienterna undersökas. Med detta menas att kommunen vid varje uppkoppling ska kunna verifiera att brandvägg och virusskydd används.

Referenser

[1] RSA Security, *What is Triple-DES?*, http://www.rsasecurity.com/rsalabs/faq/3-2-6.html, 2003-04-13 kl 11:25.

[2] SearchSecurity.com, *RSA*, http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci214273,00.html, 2003-03-12 kl 10.23.

[3] Verisign, inc., *How does a digital signature work?*, http://www.verisign.com/support/tlc/per/clientHelp/help/concepts/digsign.htm, 2003-03-12 kl 13.54.

[4] VPN Consortium, *VPN Technologies: Definitions and Requirements*, http://www.vpnc.org/vpn-technologies.pdf, 2003-03-20 kl 12.46.

[5] Bryan Bain, *Just what is L2TP anyway?*, http://www.ospmag.com/features/2000/just_what_is_l2tp_anyway.htm, 2003-03-13 kl 10.14.

[6] SearchSecurity.com, *Data Encryption standard*, http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci213893,00.html, 2003-03-15 kl 14.10.

[7] RSA security, *What is the AES?*, http://www.rsasecurity.com/rsalabs/faq/3-3-1.html, 2003-03-14 kl 12.02.

[8] RSA security, *What is a stream cipher?*, http://www.rsasecurity.com/rsalabs/faq/2-1-5.html, 2003-03-26 kl 09.54.

[9] RSA security, *What is a block cipher?*, http://www.rsasecurity.com/rsalabs/faq/2-1-4.html, 2003-03-26 kl 09.56.

[10] Lars Wallgren, *Säker identitet på nätet,* Nätverk & Kommunikation nr.5 2003

[11] Microsoft Corporation, *AH Tunnel Mode*, http://www.microsoft.com/windows2000/techinfo/reskit/en-us/cnet/cndb_ips_rtig.asp, 2003-05-07 kl 14.23.

[12] Microsoft Corporation, *ESP Tunnel Mode*, http://www.microsoft.com/windows2000/techinfo/reskit/en-us/cnet/cndb_ips_ribo.asp, 2003-05-07 kl 14.28.

[13] Microsoft Corporation, *IPSec Protocol Types*, http://www.microsoft.com/technet/prodtechnol/windowsserver2003/proddocs/entserver/sag_IPSec_Und9.asp, 2003-05-08 kl 14.58.

[14] RSA security, *What is Diffie-Hellman?*, http://www.rsasecurity.com/rsalabs/faq/3-6-1.html, 2003-05-08 kl 16.05.

A Förkortningslista

3-DES – Triple Data Encryption Standard

ACL – Access Control List

AES – Advanced Encryption Standard

AH – Authentication Header

BMAS – Border Manager Authentication Services

CA – Certification Authority

CLR – Certificate Revokation List

DES – Data Encryption Standard

DOS – Denial Of Service

DSA – Digital Signature Algorithm

ESP – Encapsulation Security Payload

GRE – Generic Routing Encapsulation

IETF – Internet Engineering Task Force

IKE – Internet Key Exchange

IP – Internet Protocol

IPsec – Internet Protocol Security

L2F – Layer 2 Forwarding

L2TP – Layer 2 Tunneling Protocol

LAN – Local Area Network

MD5 – Message Digest 5

NAT – Network Address Translation

PPP – Point to Point Protocol

PPTP – Point to Point Tunneling Protocol

RADIUS – Remote Access Dial-In User Services

RSA – Rivest Shamir Adleman

SA – Security Association

SHA – Secure Hash Algorithm

TCP – Transmission Control Protocol

UDP – User Datagram Protocol

VPN – Virtual Private Network, Virtuellt Privat Nätverk