Tekniker för filtrering av elektronisk kommunikation

kirei

Tekniker för filtrering av elektronisk kommunikation

F. Ljunggren, J. Schlyter

Innehåll

1 Introduktion

Då system kopplas samman med varandra över olika former av nätverk uppstår risker som följd av de nya kommunikationsvägarna. För att lindra dessa risker tillämpas filtrering av den kommunikation som flödar över nätverket. Filtreringen kan utformas på många olika sätt och på olika nivåer för att möta olika hotbilder. Filtreringstekni-kerna utvecklas konstant, liksom att hotbilderna blir alltmer avance-rade.

Denna PM syftar till att belysa de tekniker som traditionellt använts inom IP-baserade nätverksinfrastrukturer, och den utveck-ling som sker på området.

Målgrupp och syfte

Denna PM vänder sig till tekniskt orienterade systemarkitekter och ansvariga för upphandling och införande av system för filtrering av elektronisk kommunikation.

2 Säkerhetsarkitekturer

2.1 Introduktion till filtreringstekniker

För att lindra de risker som uppstår när system sammankopplas med varandra över olika former av nätverk tillämpas allmänt metoden att logiskt sektionera nätverken i en säkerhetsmässig zonindelning, och att kontrollera den kommunikation som flödar mellan dessa zoner. Storleken på sådana zoner kan variera från enstaka noder till en organisations hela interna nätverk.

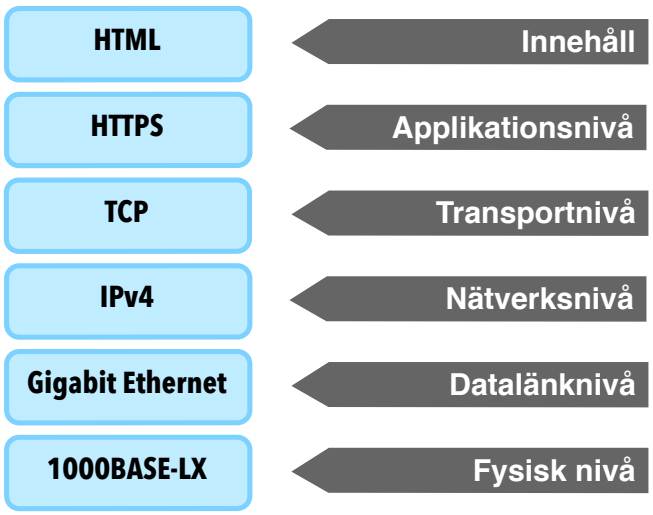
I detta zonbegrepp innefattas att de informationssystem som ingår i en viss zon på en principiell nivå tillåts kommunicera sinsemellan obehindrat. Mellan zoner upprätthålls istället ett regelverk som styr och kontrollerar de kommunikationsflöden som initieras till och från respektive zon. Sådana regelverk upprätthålls normalt i en eller flera fokalpunkter, där all trafik mellan dessa zoner tvingas passera.

Regelverket upprätthålls därvid genom olika former av filtrering. De filter som regelverket definerar kan innehålla uppgift om initierande part, mottagande part, informationsöverföringens riktning och krav på överföringsmekanismen och informationens innehåll.

Lagermodellen

Denna PM utgår ifrån det synsätt och de tekniker som bygger upp dagens IP-baserade nätverksinfrastrukturer, snarare än den mer strikta OSI-modellen.

I IP-baserade omgivningar kan nätverkskommunikation konceptuellt indelas i ett antal olika abstraktionslager (se figur 2.1), där filtrering kan förekomma inom vart och ett eller flera av dessa lager.



Figur 2.1 – Exempel på logiska lager

I denna PM används följande lagerindelning för att beskriva filtrering på de olika nivåerna:

Fysiska nivån innefattar det gränssnitt och den modulering som krävs för att överföra elektriska, elektromagnetiska eller optiska signaler mellan två kommunicerande parter.

Datalänknivån är det logiska lager som via mekanismer för dataöverföring förmedlar information mellan två parter anslutna till samma mediasegment.

Nätverksnivån sammankopplar flera mediasegment via nätverksadressering och dirigerar nätverkspaket mellan dessa segment.

Transportnivån upprättar och upprätthåller kommunikationstillstånd och tillhandahåller en kommunikationskanal mellan två kommunicerande parter anslutna till samma nätverk.

Applikationsnivån definierar protokoll och gränssnitt för den tillämpning de kommunicerande parterna använder.

Innehållet är den faktiska nyttoinformation som parterna kommunicerar.

Filtrering kan alltså förekomma på sex olika logiska nivåer i denna konceptuella lagerindelning. I det följande beskrivs dessa nivåer mer ingående.

Filtrering på fysisk nivå

På den fysiska nivån kan filtrering förekomma genom att begränsa möjligheten till överföringen av signaler. I dess enklaste form kan sådan filtrering innebära manuella handgrepp där utrustning slås på eller av, eller att kommunikationsgränssnittet kopplas in eller ur, i syfte att tillåta eller förhindra kommunikation mot ett visst system vid en viss tidpunkt.

Annan praktisk förekomst innefattar datadioder som låter de kommunicerande parterna detektera anslutning på datalänknivå, men där kommunikationen fysiskt endast kan flöda i den ena riktningen.

På fysisk nivå kan också olika former av förbindelseövervakning med försåtsskydd implementeras. Dessa bör dock inte ses som filtrerande funktioner, då de endast har för avsikt att avbryta kommunikationen helt och hållet om förbindelsens integritet påverkas.

Filtrering på datalänknivå

På datalänknivå förekommer ofta filtrering på sådant sätt att enheter anslutna till samma mediasegment i en stjärnformad nätverkstopologi hålls åtskilda genom att kommunikation endast tillåts förekomma mellan vissa fysiska eller logiska förgreningar i nätverket. Termer som *port isolation* och *private VLAN* används ofta för att beteckna sådan funktionalitet. Filtreringen kan även reglera vilka nätverksprotokoll som är tillåtna mellan olika förgreningar, eller skapa separation mellan förgreningar genom att använda teknik för virtuella lokala nätverk – *Virtual Local Area Network* (VLAN) – till exempel IEEE 802.1q.

Tillträdeskontroll – *Network Admission Control* (NAC) – kan även upprätthållas på datalänknivå, innebärande att den part som är ansluten via en viss förgrening måste identifieras och tilldelas behörighet för att kommunikationen ska tillåtas. Exempel på sådan teknik är IEEE 802.1X och 802.11i.

Det kan också förekomma dynamisk eller statisk filtrering baserat adresseringen för *Media Access Control* (MAC). Ett sådant sätt att

dynamiskt reglera MAC-adressfiltrering är att tolka ur information som kommuniceras via *Address Resolution Protocol* (ARP) eller *Neighbor Discovery Protocol* (NDP), för att sedan knyta såväl MAC-adressfiltrering som filtrering på nätverksnivån till denna information enligt ett fördefinierat regelverk. Ett sådant regelverk kan till exempel innefatta begränsningar i antalet MAC-adresser på en och samma förgrening eller försök att via förfalskade ARP-paket (*ARP-spoofing*) leda om kommunikationsströmmar i det lokala nätverket.

Funktionalitet för filtrering på datalänknivå är ofta en integrerad del i nätverksväxeln (*switch*) som kopplar samman mediasegmenten. I trådlösa miljöer motsvaras detta av nätverkets basstationer eller den centraliserade kontrollenhet som styr basstationerna, beroende på vald arkitektur.

Filtrering på nätverksnivån

På nätverksnivån sker filtrering baserat på typ av transportprotokoll samt avsändande och mottagande parts nätverksadresser. Möjligen kan regelverket för filtrering även verka på annan information som förekommer i protokollhuvudet. Det kan till exempel reglera minsta tillåten storlek på fragment, eller vissa flaggor som anger tillvalsfunktioner (*IPv4 Options* & *IPv6 Extension Headers*). Enheter kapabla att filtrera på nätverksnivån implementerar typiskt också funktionalitet för dirigering (*routing*) av paket.

Filtrering på transportnivån

Genom att kombinera filtrering på nätverksnivån med analys av information på transportnivån kan regelverk utformas som definierar vilka tjänster som ska vara åtkomliga för vilka avsändaradresser. För sessionsorienterade transportprotokoll är det också möjligt att utforma dynamisk tillståndshållande filtrering genom att analysera information som härrör till den aktuella kommunikationskanalen. Sådan filtrering kan göras mer finmaskig och därigenom tillhandahålla ett bättre skydd mot olika former av nätverksdrivna angrepp.

Filtrering på applikationsnivån

Filtrering på applikationsnivån kräver att den filtrerande logiken har stöd för att tolka och analysera det enskilda applikationsprotokollet. En sådan analys medför att filtreringen i någon mån kan säkerställa

att det är det avsedda applikationsprotokollet som kommuniceras via transportprotokollet. Det kan också vara möjligt att filtrera kommunikationen baserat på till exempel anropstyper och operationer.

Det är emellertid vanligt att applikationsprotokollet är krypterat och därmed skyddat mot insyn för de filtrerande enheterna. Det som då kan analyseras och säkerställas är de meddelanden och parametrar som kommuniceras vid upprättandet av den krypterade kanalen är i enlighet med regelverket.

Innehållsfiltrering

Filtrering baserat på innehåll kan naturligtvis endast företas då kommunikationen kan analyseras i klartext. Denna filtrering kan göras med eller utan analys av det underliggande applikationsprotokollet. Den kan göras som enkel mönstermatchning eller som en djupare analys av till exempel elektroniska signaturer eller sökande efter skadlig kod.

2.2 De filtrerande enheternas uppbyggnad och funktion

Beroende på vilken konceptuell nivå den filtrerande enheten verkar har den vanligen en utformning och uppbyggnad anpassad för dess funktion. Enheter som verkar på datalänknivån kan komma att sammankoppla många mediasegment i utkanterna av ett nätverk där prestandakraven är som högst. De krav som då ställs på dessa enheter medför ofta att filtreringen behöver göras i hårdvara i en hastighet som motsvarar linjehastigheten på det aktuella mediasegmentet. Det i sin tur leder till att filtreringsfunktionen utformas som en bitvis jämförelse av datalänkramarna. Detta hindrar dock inte att den mekanism som utformar filtret, som kan vara dynamiskt, är av en mer komplex karaktär och som finns implementerad genom mjukvara.

Även enklare tillståndslös filtrering på nätverks- och transportnivån kan på motsvarande sätt vara implementerad i den filtrerande enhetens hårdvara. Filter som dock är så dynamiska att de behöver uppdateras efter praktiskt taget varje förmedlat paket implementeras vanligen genom mjukvara. Det samma gäller djupare analys på applikations- och innehållsnivån, där den logik som krävs i sig är så komplex och föränderlig att det är opraktiskt eller alltför kostsamt att

implementera i specialiserad hårdvara.

För särskilt högpresterande tillämpningar förekommer hybridlösningar även för djupare analys av trafikflöden, där vissa delar analyseras genom mjukvara som i sin tur styr filter som är implementerade i hårdvara.

2.2.1 Terminerande filtrerande enheter

Filtrerande enheter installeras vanligen i väl valda fokalpunkter i nätverket, där trafik naturligt måste passera in och ut från den eller de säkerhetszoner den filtrerande enheten skyddar. En sådan filtrerande enhet kan dock verka på två olika grundläggande sätt: genom att endast blockera eller tillåta trafik, eller genom att även modifiera de flöden som tillåts passera.

Inom den senare nämnda gruppen finns otaliga varianter. Här görs dock ett försök att på en principiell nivå kategorisera dem baserat på den inverkan de kan förväntas ha på kommunikationen:

- Vid **terminering av transportprotokollet** fångar den filtrerande enheten upp kommunikationen och terminerar transportprotokollet i den avsedde mottagarenhetens ställe. Om kommunikationen fångas upp transparent i en fokalpunkt, kommer förloppet för avsändande part te sig som att kommunikationen i själva verket sker med den avsedde mottagaren. Den filtrerande enheten sätter då mottagarens nätverksadress i svarstrafiken som genereras. Det är även möjligt att en terminerande filtreringfunktion infogas i kommunikationsflödet genom explicit konfigurering eller genom att applikationen förses med den filtrerande enhetens nätverksadress (till exempel genom namnupplösningstjänsten). Hur detta sker saknar dock principiell betydelse för den indelning som görs här.

  I den mån den anslutning som etableras är förenlig med regelverket etablerar den filtrerande enheten i sin tur en kommunikationskanal med det aktuella transportprotokollet till den avsedde mottagaren. Detta har till följd att de båda kommunicerande parternas nätverks- och transportprotokollimplementationer (protokollstacken) inte kommunicerar direkt med varandra, utan med den filtrerande enheten. En sådan terminering skyddar de kommunicerande parterna från

varandra genom att sårbarheter i protokollstacken inte kan exploateras direkt över säkerhetszonens gränser.

- **Terminering av applikationsprotokollet** innefattar som regel även terminering av transportprotokollet. I tillägg agerar den filtrerande enheten som den tjänst som anropas, och implementerar funktionalitet för att tolka och analysera de operationer som ska vidareförmedlas eller blockeras. Detta kan även innefatta att terminera kryptografiskt transportskydd, till exempel *Transport Layer Security* (TLS) [RFC5246], vilket normalt inte låter sig göras med total transparens i applikationslagret. Det kräver i typfallet att den filtrerande enheten har åtkomst till kryptografiskt nyckelmaterial som parten eller parterna, beroende på om autentiseringen är enkelsidig eller ömsesidig, förlitar sig på.

- **Mellanlagring av innehåll** kan göras i en filtrerande enhet som del i att förmedla en transaktion från en part till en annan. Mellanlagring kan även göras för att avlasta nätverksresurser, då samma innehåll ska tillhandahållas till flera mottagare.

  Mellanlagring kan även krävas för djupare innehållsanalys. Till exempel måste äktheten hos ett elektroniskt signerat meddelande verifieras i dess helhet. Ett sådant meddelande vars äkthet inte kan verifieras, och som ska blockeras i enlighet med regelverket, får troligen heller inte delvis förmedlas till mottagaren. Det blir därför nödvändigt att mellanlagra ett sådant meddelande för att kunna utföra den kompletta analysen. Sådan mellanlagring får naturligtvis påverkan på tjänstens kvalitetsegenskaper då fördröjningarna ofrånkomligen ökar.

  Då mellanlagring sker i syfte att analysera innehållet gentemot ett regelverk, för att sedan vidareförmedla eller tillgängliggöra informationen för den avsedde mottagaren kallas denna funktion för *datasluss*.

2.3 Rätt utformat informationssäkerhetsskydd

På vilket sätt filtrering av kommunikation ska utformas och införas inom en organisation avgörs genom att analysera de hot som riktas mot verksamheten och de sårbarheter som verksamheten är behäftad

med. Genom att bedöma hotens sannolikhet att negativt påverka verksamhetens tillgångar, och vad konsekvenserna av detta riskerar bli, kan ett väl avvägt informationssäkerhetsskydd införas.

Samverkande komponenter

I detta utgör de filtrerande enheternas funktion som regel endast en del i verksamhetens informationssäkerhetsskydd. Genom att utforma säkerhetsåtgärderna så att de överlappar och skapar djupledsskydd, minskas riskerna att enskilda brister i skyddet kan medföra långtgående konsekvenser. Genom att ansvaret för upprätthållande av de olika lagren inom ett sådant djupledsskydd fördelas över olika personer eller grupperingar erhålls också en separation av arbetsuppgifter, vilket medför att det totala säkerhetsskyddet inte heller är avhängigt enskilda individer.

I de fall hotbilderna som identifieras genom riskanalysen bedöms kunna komma från en högt motiverad och tekniskt kvalificerad motståndare med stora resurser kan filtrering behöva göras i flera steg med olika tekniker, för att på så sätt skapa djupledsskydd även inom filtreringen. Detta innefattar att skapa diversitet mellan olika filtreringstekniker, så att eventuella brister i en viss implementation inte medför att säkerhetsskyddet i denna del kollapsar. Det innebär också att säkerställa att den interna uppbyggnaden i de filtrerande enheterna är konstruerad enligt en fackindelning som separerar de olika funktionerna och filtren, och därmed begränsar en eventuell uppkommen skada till den enskilda delfunktionen.

Filtrering av elektronisk kommunikation är endast en komponent i ett samverkande informationssäkerhetsskydd. Filtreringen måste kombineras med andra säkerhetsåtgärder för att bilda ett väl avvägt djupledsskydd.

Granskningsbarhet

Granskningsbarhet av regelverket är en viktig del i att skapa ett tillförlitligt skydd. Vid granskning måste det vara uppenbart om och på vilket sätt regelverket upprätthålls.

Det innefattar att utforma regelverket på ett enkelt sätt som går att överblicka. Många gånger är det möjligt att bryta ner ett mer omfat-

tande regelverk i en trädstruktur som gör att granskning av vilka regler som faktiskt är i kraft aldrig behöver innebära mer arbete än att granska ett fåtal poster i ett sådant regelverk. Granskningsbarheten kräver också att den funktion som används är utformad med detta i åtanke, vilket är en betydande skillnad mellan många enklare produkter och mer avancerade system.

Det regelverk som den filtrerande enheten upprätthåller måste vara tydligt. Vid granskning bör det kunna vara uppenbart att det inte finns några brister, snarare än att det inte synes existera några uppenbara brister.

Deterministiska fellägen

Då en filtrerande enhet slutar fungera som avsett, till exempel på grund av fel i hård- eller mjukvara eller genom överbelastning, är det viktigt att det felläge som då inträder är deterministiskt och säkert. I detta bör det vara möjligt att prioritera och välja på vilket sätt den filtrerande enheten ska verka. En naturlig väg kan vara att helt sluta vidarebefordra information, men särskilt vid överbelastningsangrepp kan det vara önskvärt att upprätthålla och prioritera vissa kritiska funktioner.

Det bör finnas en förutsebarhet i de fellägen som kan inträda då en filtreringsfunktion slutar fungera som avsett, vilket förutsätter att dessa fellägen är definierande och verifierbara.

Tillit

Tillit till produktens funktion och de tjänster kopplade till produktens livscykel är ett annat viktigt område för att kunna upprätta och upprätthålla ett tillförlitligt skydd. Att en sådan produkt fungerar så som avsett, och att de uppgraderingar och ändringar som vidtas under en produkts livscykel inte komprometterar säkerheten kräver verifierbarhet i konstruktion och i utvecklings- och underhållscykeln. Det vanligaste ramverket för att skapa sådan verifierbar tillit är via *Common Critieria* (CC). CC är en internationellt erkänd standard

betecknad ISO/IEC 15408:1999[CC], och som används vid kravställning och utvärdering av IT-säkerhetsfunktioner. Standarden används för opartisk granskning genom ackrediterade certifieringsorgan.

För att en sådan certifiering ska vara meningsfull måste emellertid den fastställda säkerhetskravställningen – *Security Target* (ST) – vara relevant i den omgivning och i det syfte produkten ska tjäna. Dessa krav kan definieras genom att innefatta en eller flera skyddsprofiler – *Protection Profile* (PP).

I den mån säkerhetskravställningen är relevant finns naturligtvis ett värde i att produkten är evaluerad enligt en nivå – *Evaluation Assurance Level* (EAL) – motsvarande den tillit verksamheten fäster vid att produkten fungerar på ett korrekt och förväntat sätt. Att en produkt genomgått en sådan evaluering på en meningsfull nivå innebär också att produkten i sig är väl dokumenterad i alla avseenden relevanta för att uppnå säkerhetskravställningen.

Produkter utformade från grunden för att genomgå en certifiering på någon av de högre nivåerna enligt Common Criteria, har som regel en genomtänkt säkerhetsarkitektur och är dokumenterade på ett sätt som gör det enklare att förstå vilka säkerhetsfunktioner produkten upprätthåller och hur de fungerar. Certifieringen innebär emellertid inte att produktens funktioner är effektiva eller att produkten håller en särskild kvalitet.

3 Filtreringstekniker

Införandet av filtreringsfunktioner kan på hög nivå sägas ha två olika huvudsakliga mål:

- dels att filtrera kommunikation i riktning mot (ingress) en säkerhetszon av högre skyddsklass i syfte att skydda de informationssystem som finns inom zonen mot nätverksdrivna angrepp,
- dels att filtrera kommunikation i riktning från (egress) en säkerhetszon av högre skyddsklass i syfte att reducera risken för informationsläckage.

I båda dessa fall är filtrering att betrakta som en *kompletterande* åtgärd. En grundprincip är att varje system inom säkerhetszonen ska ha förmåga att motstå försök till säkerhetsöverträdelser på egen hand. Att reducera risken för informationsläckage är även i mångt och mycket en personalrelaterad säkerhetsfråga, där utbildning, medvetande, noggrannhet och lojalitet är av stor betydelse. De tekniska kontrollerna kan komplettera sådana kontroller, men knappast på egen hand verka som ett heltäckande skydd.

Generellt kan sägas att filtrering är mest effektiv i ingressledet för att skydda informationssystem inom en säkerhetszon från nätverksdrivna angrepp som härrör från andra säkerhetszoner. Genom filtreringen minskas risken att sårbara gränssnitt eller tjänster exponeras mot dessa yttre zoner.

Filtrering i egressledet är många gånger ett relativt trubbigt verktyg för att säkerställa att känsliga uppgifter inte lämnar den aktuella säkerhetszonen. En datadiod kan förvisso säkerställa detta med mycket hög tillförlitlighet. Samtidigt är den meningslös om det finns andra vägar att föra ut informationen på. En överlöpare som kontrollerar en tjänst med vilken kommunikationen kan flöda fritt kan naturligtvis föra ut vilken information som helst denna vägen. Filtrering på

nätverks- eller transportnivå är därför inte meningsfull i detta syfte om inte all kommunikation begränsas till en på förhand definierad uttömmande lista av tjänster som är godtagbara och till vilka det går att fästa tillräcklig nivå av tillit till.

Filtrering kan i huvudsak sägas förekomma av två olika orsaker; dels att filtrera kommunikation i riktning mot (ingress) en säkerhetszon för att skydda de resurser som finns där; dels att filtrera kommunikation i riktning från (egress) en säkerhetszon i syfte att reducera risken för informationsläckage. I båda fallen är filtreringen endast en kompletterande säkerhetsåtgärd, som måste kombineras med andra säkerhetskontroller för att bilda ett effektivt djupledsskydd.

3.1 Fysisk filtrering

Den lägsta nivån av filtrering som kan förekomma sker på den fysiska nivån, det vill säga i princip på den kabel eller det media som förmedlar de faktiska signalerna. Ett exempel på en sådan filtreringsfunktion är en så kallad datadiod. Denna kan vara i formen av en fiberoptisk kabel där mottagare och sändare är borttagna i ena kommunikationsriktningen. Detta medför att de optiska signalerna som bär informationen fysiskt är begränsade till att enbart kunna flöda i en riktning.

I och med användandet av en datadiod och säkerställandet att trafik enbart kan kan förmedlas i en riktning kan i vissa fall zoner med högre säkerhetsklassificering på ett kontrollerat sätt kopplas samman med zoner av lägre klassificering. En datadiod kan till exempel garantera att information endast kan flöda i riktning från zonerna med lägre klassificering mot den högre klassificerade zonen, och inte i motsatt riktning.

Då en datadiod enbart tillåter signaler att passera i en riktning innebär det att många vanliga transport- och applikationsprotokoll inte fungerar, eftersom dessa kräver dubbelriktad kommunikation. De flesta förekommande system som använder datadioder torde vara av militär-, myndighets- eller totalsförsvarskaraktär med avancerad hotbild. Det är då ofta fråga om speciella system utformade för att

lösa specifika uppgifter, och som redan från början är konstruerade för att fungera genom sådana datadioder.

Det förekommer även att datadioder kompletteras med funktionalitet på högre nivåer på vardera sida om dioden, som i sin tur kan emulera de högre nivåerna. Detta krävs vanligen för att datalänkprotokoll ska kunna etableras, men är även en lösning för till exempel transportprotokollet *Transmission Control Protocol* (TCP)[RFC793].

En fördel med en datadiod är att det blir möjligt att med mycket hög tillförlitlighet och verifierbarhet säkerställa filtrets funktion, då det fysiskt kan observeras hur filtret är konstruerat och på så sätt garantera dess funktion. Genom att från början konstruera sådan datadiod med verifierbarhet i åtanke kan kommersiellt tillgängliga produkter valideras med relevanta säkerhetskrav enligt *Common Criteria* (CC) EAL7+[1].

Filtrering på den fysiska nivån, till exempel genom användning av en datadiod, är relativt ovanlig och de flesta förekommande tillämpningar torde vara av militär-, myndighets- eller totalsförsvarskaraktär med avancerad hotbild.

3.2 Filtrering på datalänknivå

På liknande sätt som trafik kan filtreras på högre nivåer i protokollstacken kan även viss filtrering göras på datalänknivå. Ett vanligt förekommande datalänknivåprotokoll är Ethernet (IEEE 802.1), där filtrering ofta sker som ett resultat av indelning av det lokala nätverket i flera logiska nätverkssegment, så kallade *Virtual Local Area Network* (VLAN) (IEEE 802.1q). Tekniken används för att upprätta logiska säkerhetszoner inom en och samma fysiska nätverksinfrastruktur och för att koncentrera informationsflöden till fokalpunkter där de kan filtreras.

Genom att flera säkerhetszoner kan hanteras inom samma utrustning och kablar uppnås kostnadsfördelar. Samtidigt kan hotbilden kräva djupledsskydd, så att flera samverkande åtgärder upprätthåller

[1] Fort Fox Hardware Data Diode, version FFHDD2+

den separation mellan säkerhetszoner som anses nödvändig. Det kan kräva att säkerhetszoner med vitt skilda skyddsbehov ändå behöver separeras på den fysiska nivån, medan säkerhetszoner med samma klassificering kan separeras logiskt genom filtrering på datalänknivån.

Denna filtrering av datalänkramar och separation av kommunikationsflöden upprätthålls av den nätverksutrustning som bygger upp nätverket, det vill säga switchar i nätverkets ryggrad och i de olika distributionslagren. För att sedan koppla samman de olika zonerna på nätverksnivån via ett system för filtrering av kommunikation krävs förstås att samma teknik finns implementerad i de nätverkselement som ska göra denna filtrering.

Produkter för nätverksseparation

För att dela in det fysiska nätverket i de logiska förgreningar som ska utgöra de säkerhetszoner vars kommunikation sedan ska filtreras används funktionalitet i de växlar (*switch*) som bygger upp nätverket på datalänknivå. De enklaste produkterna implementerar IEEE 802.1Q för indelning i virtuella lokala nätverk (VLAN).

I mer avancerade produkter finns funktioner för att inom ett och samma lokala nätverk separera varje fysisk förgrening så att dessa endast kan kommunicera med vissa utpekade noder (ofta kallat *private VLAN* eller *port isolation*). Sådan separation är särskilt användbar då ändnoderna på nätverket inte är betrodda av varandra, eller då förgreningarna sträcker sig utanför organisationens fysiska skalskydd. Dessa produkter förekommer därför ofta i områdesnät eller samlokaliseringsanläggningar med gemensam nätverksinfrastruktur.

Andra produkter med tydlig inriktning på användarmiljöer kan implementera IEEE 802.1X och utefter användarens identitet styra vilket virtuellt nätverk som denne ska anslutas till. Tekniken kan naturligtvis även fungera i servermiljöer, men där torde den vara mer ovanlig då dess anslutningar ofta är tämligen väl skyddade och med få förflyttningar.

Tekniker för att kringgå filtrering på datalänknivå

Filtrering på datalänknivå som syftar till att skapa ett skydd mellan olika förgreningar inom ett nätverkssegment sker genom ett antal

samverkande funktioner. Ett filter som blockerar vissa typer av nätverksprotokoll kan vara ett effektivt skydd mot att utnyttja nätverksprotokoll som inte är i enlighet med regelverket, och på så sätt skydda mot nätverksdrivna angrepp. Filtreringen av nätverksprotokoll är dock verkningslös om motståndaren kontrollerar båda enheterna mellan vilken kommunikationen sker. Den filtrerande enheten kan inte på datalänknivå avgöra vilket faktiskt nätverksprotokoll som transporteras i ramarna, och det är därför möjligt att transportera vilket nätverksprotokoll som helst i en ram som anger en tillåten protokolltyp.

Genom att filtrera datalänknivåramarna baserat på avsändande eller mottagande MAC-adress kan kommunikationen i någon mån styras inom ett nätverkssegment. Det förutsätter emellertid att en motståndaren inte har kontroll över datalänklagret i en nod som används för kommunikationen, då MAC-adressen trivialt kan sättas till godtyckligt värde och därmed förfalska datalänkramar så att de förefaller komma från en annan värddator. Portisoleringstekniker implementerar därför ett statiskt skydd genom en filtertabell kallad *Content Addressable Memory* (CAM). Tabellen innehåller bindningar mellan de dynamiskt inlärda MAC-adresserna och de fysiska portarna. Filtren verkar alltså på den del som innehåller uppgifter om fysisk port, snarare än de dynamiskt inlärda MAC-adresserna. Detta har till följd att portisoleringstekniken är ett starkare men mer statiskt skydd än MAC-adressfiltrering.

MAC-adressfiltrering som bygger på avsändande adress kan betraktas som ett medel för att förhindra enkla misstag. MAC-adresser representeras i många olika datalänklagertekniker som en 48 bitar lång adress enligt IEEE EUI-48. Att kringgå ett sådant filter innebär att gissa en i regelverket giltig adress. Som tidigare angavs är en MAC-adress 48 bitar lång. I praktiken kan betydande delar av adressutrymmet uteslutas helt eller betraktas som osannolika av en angripare, vilket gör det möjligt att maskinellt prova olika adresser till dess att en giltig hittas. Sökandet underlättas ytterligare om tillverkaren av det tillåtna datalänkgränssnittet är känt, då tillverkare ansöker om och blir tilldelade block om 24-bitar stora adressrymder, varvid de första 24-bitarna är offentliga och kända på förhand.

En vanligt förekommande form av filtrering på datalänknivå, som också utgör en viktig del i att skapa de fokalpunkter som krävs för filtrering på de högre nivåerna, är genom inrättandet av virtuella lokala nätverk (VLAN) där trafik från olika förgreningar märks upp och hålls logiskt åtskilda som vore de anslutna till separata mediasegment.

3.3 Tillståndslös paketfiltrering

Den enklaste formen av filtrering på nätverksnivå är av typen tillståndslös paketfiltrering. Med tillståndslös menas att den enhet som tillämpar filtreringen inte analyserar om IP-paketet som passerar filtret har någon association till redan etablerade dataströmmar eller inte. Varje paket analyseras och hanteras var för sig, och på grundval av det regelverk som är definierat avgörs om paketet ska vidareförmedlas eller stoppas. Tillståndslös paketfiltrering baserar i huvudsak filteringsbeslut på information som förekommer i IP-paketets protokollhuvud, och behöver därmed inte behandla paketet mer ingående än så. Detta medför att det åtgår förhållandevis begränsade resurser att tillämpa regelverket och fatta filtreringsbeslut.

En av nackdelarna med tillståndslösa paketfilter är att de ofta har begränsad verkan på nätverksdrivna angrepp som utnyttjar sårbarheter i protokollstacken, till exempel genom fragmentering av paket eller genom förfalskade avsändaradresser (*IP spoofing*). Regelverket kan inte heller göras mer finmaskigt än att all tänkbar svarstrafik måste tillåtas statiskt.

En tillståndslös filtrering kan dock även verka över information i transportprotokollets huvud, till exempel portnummer och protokollflaggor. På så sätt kan även kommunikationens initieringsriktning styras för till exempel TCP.

Tillståndslös paketfiltrering förekommer ofta vid enklare former av trafikstyrning och tillgångskontroll. Exempel innefattar enklare listor över tillåtna nätverksadresser som upprätthålls av routrar och andra nätverkselement, där viss typ av trafik ska grovfiltreras. Tillståndslösa paketfilter används i sådana nätverkselement både för transiterande och terminerande kommunikation. För transiterande kommunikation tillämpas paketfilter för att avgöra om noden ska

vidarebefordra eller blockera viss typ av kommunikation. För terminerande kommunikation, till exempel mot enhetens egna administrativa gränssnitt, används paketfilter för att avgöra om kommunikationen ska förmedlas vidare till nodens applikationssystem eller om trafiken ska blockeras.

Tekniker för att kringgå tillståndslös filtrering

Regelverk som ska upprätthållas genom tillståndslös filtrering måste statiskt tillåta alla tänkbara avsändaradresser och portnummer för svarstrafik. Om trafik endast ska vara tillåten i en riktning över zongränsen, betyder det att ett stort antal portar behöver vara åtkomliga i motsatt riktning. Om den tillståndslösa filtreringen inte också analyserar information på transportprotokollnivå blir det trivialt att initiera kommunikation även i denna riktning med den begränsningen att endast svarsportar kan adresseras.

Sådan tillståndslös filtrering har heller inte förmåga att motstå försök att kommunicera med förfalskade avsändaradresser och portnummer, då dessa inte kan jämföras med ett etablerat kommunikationstillstånd. Det innebär att en motståndare som känner till (eller kan gissa) en giltig avsändare kan sända paket i motsatt riktning av vad filtret anger, utan att känna till uppgifter om etablerade kommunikationstillstånd – till exempel parametrar för TCP.

Andra metoder att kringgå icke-tillståndshållande filtrering innefattar att fragmentera paket på ett sådant sätt att då det tas emot av mottagaren skriver över vitala delar av pakethuvudet. Denna metod innefattar att först sända ett paket som är i enlighet med den filtrerande enhetens regelverk, för att sedan därpå sända ett fragment vars inledning positioneras direkt på det föregående paketets protokollhuvud så att det sammanfogade paketet då det tolkas i den mottagande värddatorn får en annan betydelse än vid filtreringstillfället. Dessa angreppstekniker kan enkelt hanteras genom att fragment med orimligt kort inledning (*offset*) blockeras oavsett mottagare eller avsändare.

Syftet med en sådan fragmentering kan emellertid också vara att kringgå filtrering på de högre nivåerna, och på detta sätt skicka en ström av paket som – då de sätts samman – bildar skadlig kod. En filtrerande enhet som ska verka på nätverksnivån eller högre måste därför ha förmåga att sätta samman samtliga fragment av

ett datagram innan detta vidarebefordras till mottagaren. Principen gäller för filtrering både i ingressledet (för att skydda informationssystem) och i egressledet (för att förhindra informationsläckage).

Den mest uppenbara metoden att kringgå filtrering torde annars vara att använda tunnelteknik, där en informationsöverföring innesluts i ett applikations- eller transportnivåprotokoll som tillåts av regelverket. Genom att använda totalsträckskryptering (*end-to-end*) blir det i praktiken omöjligt att avgöra vilken information som överförs eller vilket applikationsprotokoll som används för överföringen. Tunneltekniken är mest effektiv i egressledet, där den kan vara mycket svår att upptäcka och blockera om ansträngningar görs för att maskera den. För att utnyttja tunnelteknik i ingressledet krävs dock att en tunneltjänst är tillgänglig inom den säkerhetszon som angriparen önskar initiera tunneln till.

Tillståndslös paketfiltrering är mest effektiv att införa i ingressledet i nätverkets routrar. Här kan en grov filtrering göras för att blockera felaktiga avsändaradresser genom att införa ett regelverk som speglar inversen av routingtabellen (*reverse path filtering*), samt för att skydda routrarnas egna administrativa gränssnitt.

3.4 Tillståndshållande paketfiltrering

Den vidareutvecklade formen av filtreringsfunktion på nätverks- och transportnivå är av typen tillståndshållande paketfiltrering, även kallad tredje generationens brandvägg eller *Stateful Packet Inspection* (SPI). Som benämningen antyder är den största skillnaden att denna typ av paketfilter inte bara analyserar varje enskilt IP-paket för sig, utan även kontrollerar paketet i förhållande till etablerade kommunikationsflöden. Vanligen tillämpas regelverk som endast tillåter svarstrafik om denna kan associeras med ett befintligt kommunikationsflöde som initierats i en riktning förenlig med regelverket.

Varje nytt flöde som initieras genom den filtrerade enheten analyseras vid upprättandet, och filter som reglerar just det flöde som de båda kommunicerande parterna använder inrättas dynamiskt. Det dynamiskt inrättade filtret kan tillämpa en striktare kontroll baserat på samtliga de uppgifter som finns tillgängliga i såväl nätverkspro-

tokollets som transportprotokollets huvuden. Filtret är verksamt så länge kommunikationen fortgår. Då parterna avslutar kommunikationen, eller att viss tid förflutit utan att någon kommunikation skett, tas det dynamiska filtret bort.

Beroende på vilket transportprotokoll som används har en tillståndshållande filtrerande enhet olika förmåga att identifiera ett trafikflöde. För sessionsorienterade protokoll, till exempel TCP, kan den filtrerande enheten utifrån trevägshandskakningen skapa och upprätthålla flödesidentifierare som innehåller IP-adresser, portnummer, protokollflaggor, sekvensnummer, och så vidare. TCP har även inbyggda funktioner för att hålla sessionen vid liv, så kallad *keep-alive*, vilket gör det enkelt för den filtrerande enheten att också avgöra om flödet fortfarande är aktivt. När TCP avslutar en session kan även den filtrerande enheten ta bort flödet från tillståndstabellen och frigöra resurserna.

Generellt kan sägas att det administrativa regelverket styr hur flöden får initieras, och den tillståndshållande filtrerande enhetens funktioner påverkar sedan i tillägg till detta hur svarstrafik tillhörande redan etablerade flöden kan styras.

De parameteruppsättningar som definierar ett TCP-flöde kan normalt heller inte enkelt gissas, vilket ger ett effektivt skydd mot förfalskade svarspaket (*IP spoofing*). För tillståndslösa transportprotokoll, till exempel *User Datagram Protocol* (UDP), är tillståndet svagare definierat. UDP innefattar inte någon trevägshandskakning, och inte heller någon sekvensnumrering eller flaggor i protokollhuvudet som anger hur ett paket förhåller sig till en pågående kommunikationsström. En filtrerande enhet kan i detta fall endast associera ett kommunikationsflöde till avsändande och mottagande parts adresser samt portnummer. Enda sättet att avgöra när ett kommunikationsflöde är inaktivt är att inrätta en tidsgräns för hur länge tillståndet ska vara aktivt. Applikationer som upprätthåller långvariga kommunikationskanaler över UDP behöver därför ofta implementera någon form av regelbunden kommunikation i form av *keep-alive*.

Den tillståndshållande filtreringen är mer resurskrävande än motsvarande tillståndslös filtrering, vilket får påverkan på den filtrerande enhetens prestanda. Den mer komplexa tillståndshållande filtreringen implementeras ofta i generisk hårdvara, men kan i högprestandatillämpningar använda specialiserad hårdvara (FPGA/ASIC) i dataplanet. Förekomst av mycket hög belastning, till

exempel som följd av tillgänglighetsangrepp, kan komma att ta betydande resurser i anspråk från den filtrerande enheten. Genom översvämning av till exempel DNS-frågor kommer miljontals tillstånd kunna etableras på kort tid, och alla dessa tillstånd kommer inte kunna hanteras. Då är det viktigt att det felläge som inträder är deterministiskt och säkert, och att tillgängliga resurser avdelas att upprätthålla de viktigaste tjänsternas tillgänglighet.

De flesta idag på marknaden förekommande brandväggsprodukter bygger i någon form på den tillståndshållande filtreringens principer. Skillnader kan bland annat förekomma i förmåga att knyta kommunikationstillstånd till högre nivåer, det vill säga applikation och innehåll. En tillståndshållande filtrerande enhet med applikationslogik för till exempel *Domain Name System* (DNS) [RFC1034] kan knyta sådana kommunikationstillstånd till DNS-protokollets transaktionsidentifierare. Det finns även tillämpningar där kommunikationstillstånd etableras som en följd av varandra i olika riktningar med dynamiska portar, och där innehållet i applikationsprotokollet kan behöva analyseras för att tillåta initiering av dessa kommunikationskanaler. Exempel innefattar *Session Initiation Protocol* (SIP) [RFC3261], där signaleringen ger upphov till nya kommunikationstillstånd, eller *File Transfer Protocol* (FTP) [RFC959] där dataöverföringskanalen är separat från kommandokanalen.

Produkter för tillståndshållande filtrering

I det som i vardagligt tal och som enligt branschstandard benämns som en brandvägg innefattas funktioner för finmaskig tillståndshållande filtrering med förmåga att sätta samman fragmenterade segment och förhindra olika former av resursuttömningsangrepp på transportnivån[2]. Brandväggen gör vanligen dirigering av kommunikation på nätverksnivån, inklusive adress- och portöversättningsfunktioner (NAT/PAT).

Vidare implementerar de vanligen olika former av tunnelmekanismer, till exempel *General Routing Encapsulation* (GRE) [RFC1701] och *Internet Protocol Security* (IPsec) [RFC4301], vars funktioner syftar

[2]Till exempel angrepp som syftar till att lämna stora mängder anslutningar i halvöppet tillstånd (*SYN-flood*).

till att knyta samman nätverk via virtuella länkar, så kallade virtuella privata nätverk – *Virtual Private Network* (VPN).

Mer avancerade brandväggslösningar har funktionalitet för att använda flera noder i en redundant installation och centralt objektorienterat administrationgränssnitt för att styra flera sådana installationer som ett integrerat filtreringssystem. Ofta finns även viss applikationsnivåfiltrering för vanliga applikationsprotokoll, vilket i detta sammanhang vanligen benämns *Application Layer Gateway* (ALG). Även de enklaste brandväggar implementerar dock funktionalitet för att dynamiskt anpassa filtreringen så att associerade kommunikationskanaler kan upprättas för till exempel protokoll som FTP, H.323 och SIP [RFC3261].

Moderna brandväggar implementerar tillståndshållande filtrering som en grundläggande del i de filtreringsfunktioner de erbjuder. Funktioner för adress- och portöversättning ingår vanligen också, men bör i detta sammanhang betraktas som en funktion för dirigiering av paket, snarare än en filtreringsfunktion.

3.5 Applikationsnivåfiltrering

En funktion för applikationsnivåfiltrering upprätthåller ett regelverk som styr kommunikation som sker på applikationslagret i protokollstacken. Applikationsnivåfiltrering kan implementeras i en förmedlingsnod i nätverket som kanske även verkar på underliggande protokollnivåer. Det är emellertid vanligt att den särskilda logik som krävs, och som vanligen också är tämligen komplex, implementeras i en särskild funktion dit trafiken styrs på ett eller annat sätt. Det kan göras genom att en förmedlingsnod som till exempel upprätthåller tillståndshållande filtrering separerar ut applikationsinformationen och förmedlar den via *Internet Content Adaptation Protocol* (ICAP) [RFC3507] till den applikationsnivåfiltrerande funktionen.

En applikationsnivåfiltrerande enhet som ska upprätthålla ett regelverk för en viss typ av tillämpning måste ha förmåga att tolka det specifika applikationsprotokollet och utverka filtreringsbeslut baserat på dess innehåll. Till exempel kan en applikationsnivåfiltrerande enhet implementera funktionalitet för att filtera DNS-protokollet.

Regelverket kan till exempel styra att endast tillåta anrop och svar som uppfyller de aktuella specifikationerna för protokollet.

Detta har till följd att funktionen i den applikationsnivåfiltrerande enheten ständigt måste följa med i den tekniska utveckling som sker av de protokoll och tjänster som omfattas av regelverket. Om den filtrerande enheten inte stöder nya funktioner i de applikationsnivåprotokoll som ska förmedlas finns annars risk att även legitim trafik förhindras att passera, eller att nya hot som uppkommer inte kan hanteras optimalt och att det stipulerade regelverket inte fullt ut kan upprätthållas av brandväggen.

Vissa applikationsprotokoll är betydligt mer komplexa än andra, vilket ställer högre krav på den filtrerande enheten att kunna förmedla dessa på rätt sätt. Ett exempel på ett särskilt komplext applikationsprotokoll är SIP [RFC3261]. En applikationsnivåfiltrerande enhet kan inte enbart analysera innehållet i SIP-huvudet utan måste även ta hänsyn till de mediarelaterade trafikflödena som SIP-signaleringen är menad att etablera. Dessa beskrivs ofta i underliggande protokoll, till exempel *Session Description Protocol* (SDP), som i sin tur transporteras av SIP.

På liknande sätt finns det en uppsjö av olika applikationsprotokoll och tjänster som använder sig av *Hypertext Transfer Protocol* (HTTP) [RFC7230]. Det räcker inte för en applikationsnivåfiltrerande enhet att ha förmåga att tolka HTTP, den måste även kunna tolka innehåll för att vara meningsfull, till exempel i form av *Hypertext Markup Language* (HTML) och JavaScript.

Sammantaget kan sägas att applikationsnivåfiltreringen endast kan verka effektivt över de delar av kommunikation som kan tolkas. Det fordrar också att syftet med applikationsnivåfiltreringen är definierat, och vad som (mot bakgrund av de identifierade riskerna) är tillåten eller inte tillåten kommunikation.

Om kommunikationen är krypterad eller på annat sätt skyddad under transport så att mellanliggande noder inte kan tolka innehållet reduceras nyttan med applikationsnivåfiltreringen. I många fall måste totalsträckskryptering brytas för att en applikationsnivåfiltrerande förmedlingsnod ska kunna styra kommunikationen och upprätthålla regelverket, vilket i sin tur kan ge upphov till andra risker.

Syftet med applikationsnivåfiltreringen måste vara tydligt identifierat och omsatt i ett effektivt regelverk för att filtreringen ska vara meningsfull. Om syftet inte är uttalat, och det inte finns någon identifierad risk som måste lindras, kommer applikationsnivåfiltreringen sannolikt endast medföra en rad nackdelar.

I de fall kryptografiskt transportskydd (*end-to-end*) måste brytas för att möjliggöra applikationsnivåfiltrering uppstår som regel ett antal andra tämligen väsentliga risker. Det bör i sådana fall även övervägas om alternativa mindre problematiska och möjligen mer effektiva risklindrande kontroller är lämpligare att införa i andra delar av IT-miljön än genom applikationsnivåfiltrering av kommunikation.

3.6 Innehållsfiltrering

Produkter för innehållsfiltrering finns i en mängd olika skepnader med olika förmågor att lösa mer eller mindre specifika uppgifter. De kan indelas i tre olika huvudkategorier:

- *Data Loss Prevention* (DLP) som verkar i egress-ledet, och som syftar till att minska risken för infomationsläckage från en säkerhetszon;
- *Intrution Prevention System* (IPS) som verkar i ingress-ledet, och som syftar till att minska risken för intrång i interna informationssystem; samt
- Filter för användarinteraktion, som syftar till att minska risken för att säkerheten i klientmiljön ska kompromettetras som följd av användares handlingar.

Samtliga kategorier av system behöver utformas att verka som en *datasluss*, innebärande att hela den informationsmängd som ska överföras behöver analyseras i sin helhet innan beslut kan tas att blockera eller vidareförmedla.

Data Loss Prevention (DLP)

För filtreringssystem vars syfte är att förhindra att skyddsvärda uppgifter förmedlas via filtreringsfunktionen krävs att dessa

uppgifter på något sätt kan identifieras. Ett system som syftar till att förhindra sådant informationsläckage arbetar därför vanligen i två steg:

- Ett steg som omsätter ett definierat regelverk till kriterier för att blockera eller vidareförmedla viss information. I detta steg kan information märkas upp med elektroniska signaturer eller att det skapas fingeravtryck av information (eller fragment av information) som ska blockeras (svartlista) eller kan vidareförmedlas (vitlista).

- Ett steg som analyserar en informationsöverföring och jämför den med de fastställda kriterierna för att därefter besluta om överföringen ska blockeras eller kan vidareförmedlas.

Komplexiteten i ett system som syftar till att motverka informationsläckage ligger i det första steget där all information måste klassificeras som antingen tillåten att överföra eller icke-tillåten att överföra. Huruvida det är lämpligt att tillämpa en vitlista, det vill säga uttömmande kriterier för att tillåta en överföring, eller en svartlista, det vill säga uttömmande kriterier för att blockera en överföring, beror på den hotbild som anses existera.

De flesta kommersiella produkter som saluförs på marknaden bygger vanligen på teknik för att skapa en svartlista över känsliga informationstillgångar. Uppgifter om dessa informationstillgångar kan inhämtas regelbundet och kriterier skapas utifrån dessa.

Att på omvänt sätt skapa vitlista över den information som får överföras har till främsta syfte att möjliggöra att information som på något stadie genomgått en godkännandeprocess kan vidareförmedlas.

Att kontrollera riskerna för informationsläckage genom innehållsfiltrering kräver ett effektivt system för klassificering och märkning av information. I de IT-miljöer som förekommer idag saknas vanligen tekniskt stöd för sådan klassificering och märkning, varvid skydd mot informationsläckage genom innehållsfiltrering blir ett tämligen trubbigt instrument med osäker grad av skydd.

Intrusion Prevention System (IPS)

Ett system som syftar till att motverka intrångsförsök grundar filtreringen på kända angreppsmönster och verkar vanligen i förening med den tillståndshållande filtreringen på transport- och nätverksnivån. En nätverksadress som uppvisar ett fientligt beteende kan blockeras helt från att kommunicera med de informationssystem som ska skyddas. Samtidigt kan larm skapas för att påkalla uppmärksamhet kring de omständigheter som gör gällande att ett angrepp kan ha iscensatts och eskalera händelsen till den instans som kan vidta rätt åtgärder.

Ett system som syftar till att upptäcka och avbryta möjliga intrångsförsök kräver ofta betydande anpassningar i konfiguration för att inte generera alltför omfattande mängder falsklarm och för att inte blockera legitim kommunikation. Samtidigt är dess verkningsgrad beroende av att angreppsmönstret är känt och går att särskilda från vad som annars förfaller vara normal kommunikation.

För en organisation med en avancerad hotbild kan ett kommersiellt IPS-system antas ha begränsad verkan, och skyddsåtgärderna bör därför troligen koncentreras till de system där sårbarheterna riskerar att uppstå.

Filter för användarinteraktion

Filter för användarinteraktion syftar till att styra den information som flödar i riktning mot användaren. Filtreringen kan vidtas av effektivitetsskäl, till exempel i syfte att sortera bort skräppost eller för att blockera resurser på Internet som inte är arbetsrelaterade. Det kan också syfta till att begränsa vådligt användarbeteende, som till exempel att upprätta tunnlar mellan olika säkerhetszoner som därvid kortsluter filtreringsmekanismerna eller begränsa en viss tillämpnings funktionalitet i applikationsprotokollet. Funktionerna kan även söka i informationsöverföringar efter skadlig kod, et cetera.

För att denna typ av filtrering ska bli meningsfull måste i de flesta fall transportskyddet brytas, vilket i praktiken innebär att den filtrerande enheten fullbordar en janusattack på den av användaren initierade sessionen. Detta har till följd att användaren inte längre kan verifiera motpartens identitet, då transportskyddet konsekvent termi-

neras i filtreringssystemet. Vidare kan inte heller motparten verifiera användarens identitet, i de fall ömsesidig autentisering krävs.

Då den här typ av filtrering tillämpas för filtrering av kommunikation mot Internet förloras även möjligheten att indikera organisationsnamn via så kallade EV-certifikat. Det finns också risk att vissa webbplatser blir onåbara genom användande av certifikatlåsning (*certificate pinning*), innebärande att klientens webbläsare sparar ett betrott certifikat vid ett tillfälle, och accepterar sedan inte ett annat certifikat för samma webbplats vid ett senare tillfälle. Andra typer av certifikatvalidering som till exempel förankring av tillitskedjan i DNS vi DANE [RFC6698] kommer inte heller fungera.

På det hela taget är tekniker för filtering av användarinteraktion ofta problematiska, inte bara från strikt teknisk synvinkel. De innebär att uppgifter som ska vara krypterade avkrypteras och analyseras utanför de system som kommunicerar dem. Det nyckelmaterial som det filtrerande system kräver åtkomst till blir dessutom av ytterst känslig karaktär, och måste skyddas på ett tillfredställande sätt.

3.7 Nästa generations brandvägg

Next Generation Firewall (NGFW) är en marknadsföringsterm som avser filtreringssystem där filtrering från nätverksnivån och upp till innehållsnivån kopplats samman i en integrerad plattform. Centralt kan sägas är förmågan att identifiera tillämpningar utifrån det applikationsprotokoll som används, snarare än utifrån transportprotokollets portnummer. Vidare avses plattformen ha förmåga att identifiera de flesta typer applikationsprotokoll, företrädesvis för att förhindra olika former av tunneltekniker och annat missbruk. En annan term som förekommer är *Unified Threat Management* (UTM), som i praktiken är ekvivalent med NGFW. De tillverkare som saluför produkter enligt termen NGFW gör ofta gällande att det finns en skillnad i hur tätt de olika funktionerna integrerats och att det finns en skillnad i prestanda, kvalitet och produktens mognadsgrad. Dessa delar kan dock anses vara tämligen subjektiva, och konceptuellt bör de båda termerna utifrån detta resonemang betraktas som ekvivalenta.

Produkter som bygger på NGFW/UTM gör det relativt enkelt att införa avancerade filtreringsfunktioner genom en integrerad plattform, snarare än att på egen hand integrera specialiserade komponenter. Nackdelen är att en sådan NGFW/UTM-plattform är utformad utefter en generell hotbild, är synnerligen komplicerad i dess uppbyggnad och fordrar kontinuerlig utveckling och uppdatering av dess funktion för att i någon mån kunna följa utvecklingen av applikationsprotokoll och tillämpningar. Vidare följer att icke-nödvändig funktionalitet av naturen inte behöver finnas, och medför istället en större angreppsyta och högre risk än vad som kanske erfordras för att lösa en viss uppgift.

För organisationer med avancerad hotbild krävs sannolikt mer specialiserade filtreringssystem bestående i diskreta komponenter med högre tillitsgrad, verifierbarhet och granskningsbarhet. Att utforma sådana anpassade filtreringssystem fordrar emellertid högre kompetens. Produkter enligt NGFW/UTM-konceptet kan ofta införas av generalister med allmän kunskap kring nätverkskommunikation, samtidigt som ett större ansvar faller på leverantören av produkten att vidmakthålla dess funktion genom kontinuerlig utveckling av applikationslogiken.

4 Filtreringssystemens uppbyggnad

En systemlösning för upprätthållande av ett regelverk för filtrering består vanligen av ett antal samverkande komponenter med olika uppgifter. I högt specialiserade filtreringssystem kan dessa moduler vara helt separerade, medan de i mindre mindre systemlösningar ofta är integrerade som en enhet.

Kontrollplan

Kontrollplanet styr filtreringen och innehåller alla de stödkomponenter som krävs för att styra regelverk, loggföra händelser, uppdatera vägvalstabeller och annan konfiguration, med mera.

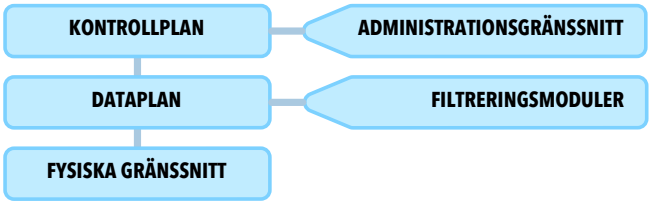
Funktioner som normalt implementeras i kontrollplanet innefattar:

- konfigurationsstyrning;
- styrning av vägval, till exempel genom dynamiska vägvalsprotokoll;
- övervakningsfunktioner;
- funktioner för loggföring; samt
- tidssynkronisering.

Om uppdelningen mellan kontroll- och dataplan är strikt kan flera redundanta kontrollplan samexistera i en och samma systemlösning. De två kontrollplanen är i sådana fall ofta konfigurerade i ett aktivt/passivt förhållande, där det för närvarande aktiva kontrollplanet är det som vid varje givet tillfälle styr dataplanet.

Skydd av kommunikation mot kontrollplanet hanteras i många fall helt skilt från övrig trafikfiltrering. Konceptuellt kan en sådan uppbyggnad betraktas som att kontrollplanet har en egen filtreringsfunktion för inkommande trafik från dataplanet samt från de till kontrollplanet direkt anslutna kommunikationsgränssnitten. Denna funktion benämns vanligen *Control Plane Policing*.

Vanligen baseras kontrollplanet i ett modernt system för filtrering på ett generiskt operativsystem av UNIX-typ (till exempel Linux eller FreeBSD) och som exekveras i en standardarkitektur, till exempel x86, ARM eller PowerPC.



Figur 4.1 – Kontroll- och dataplan

Dataplan

Dataplanet är den del av den filtrerande enheten som tar emot, hanterar och vidarebefordrar trafik. Trafik adresserat till den filtrerande enheten självt måste i många fall hanteras av kontrollplanet, även om vissa typer av trafik, till exempel vissa ICMP-meddelanden, kan hanteras redan i dataplanet.

Dataplanet kan realiseras på en mängd olika sätt. I mindre integrerade filtrerande enheter är det vanligt att en och samma processor används av både kontroll- och dataplanet, oftast genom att filtreringsfunktionen överlåts till kärnan i det generella operativsystem som den filtrerande enheten baseras på. I större och mer specialiserade systemlösningar används ofta diskreta resurser för dataplanet. Detta kan vara dedicerade CPU-kärnor eller helt separat hårdvara.

Dedicerade CPU-kärnor har fördelen att de är enkla att programmera och ger god prestanda till lågt pris. För högre prestanda används istället programmerbar hårdvara, *Field-Programmable Gate Array* (FPGA), eller helt specialkonstruerade integrerade kretsar,

Application-Specific Integrated Circuit (ASIC).

4.1 Filtrering på applikationsnivå och av innehåll

Applikationsnivåfiltrering kräver i princip alltid att dataströmmen termineras och återupprättas i den filtrerande enheten, men kan även förekomma som transparent funktion där den filtrerande enheten buffrar och i denna buffert återskapar innehållet för analys. Då överföringen analyserats töms successivt bufferten för att slutföra överföringen till den mottagande parten.

Oavsett om dataströmmen ska termineras på transportnivå eller bara passivt analyseras, behövs i de allra flesta fall en nätverksstack. Denna kan antingen vara implementerad i kärnan i det operativsystem den filtrerande enheten baseras på, alternativt hanteras av dataplanet separat. Det senare är att föredra då det minskar konsekvenserna av en sårbarhet i nätverksstacken. Genom att utnyttja operativsystemsfunktioner för att separera och avgränsa varje sådan process från övriga delar av systemet minskas risken att en sårbarhet får fatal påverkan på säkerheten i filtreringssystemet. Detta gäller inte minst i situationer med försök till resursuttömning.

När den filtrerande enheten har tillgång till den dataström som ska filtreras kan själva filtreringen ske antingen lokalt i den filtrerande enheten eller genom att dataströmmen skickas vidare till särskild analysfunktion. Detta har berörts närmare i kapitel 3.5.

4.2 Logiska nätverkselement

Som alternativ till att bygga flera fysiska system för trafikfiltrering kan olika former av logiska nätverkselement utföra samma funktioner i en virtualiserad eller på annat sätt uppdelad omgivning. Dessa nätverkselement kan realiseras på en mängd olika sätt, varav några av de vanligast förekommande presenteras nedan.

4.2.1 Virtuella system

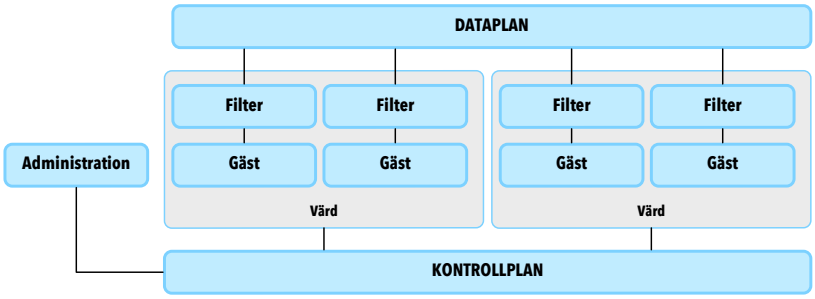
Med logiska nätverkselement i en och samma instans menas ett filtrerande system som internt är uppdelat i flera nätverkselement med egna logiska kontroll- och dataplan. Varje logiskt nätverkselement

har sin egen vägvalstabell, *Virtual Routing and Forwarding* (VRF), och uppsättning filtreringsregler. Ofta finns möjlighet att delegera ansvar för administrationen av de logiska elementen till olika administratörsgrupper.

Flera virtuella instanser av filtrerande system kan också realiseras i en virtualiseringsplattform (*hypervisor*). Till skilland från flera logiska nätverkselement i en och samma instans, exekveras här de filtrerande systemen var för sig och delar endast på den underliggande fysiska hårdvaran, inklusive tillhörande nätverksgränssnitt.

4.2.2 Mjukvarudefinierade nätverk

Mjukvarudefinierade nätverk, *Software-defined Networking* (SDN), är en utökning av och komplement till virtualiseringsplattformens traditionella funktioner, som i många fall kan användas för att implementera enklare filtreringsfunktioner. I det mjukvarudefinierade nätverket skapas förgreningar mellan de virtuella värddatorer (gäster) som realiseras i virtualiseringsplattformen, och det för virtualiseringsplattformen gemensamma regelverket upprätthålls i de värdar, *hypervisors*, som hanterar de virtuella värddatorerna. Samtidigt kan administration av regelverk ske via ett centralt administrationsverktyg.



Figur 4.2 – Filtrering i SDN

Regelverk för trafikfiltrering inom ett mjukvarudefinerat nätverk gör det enkelt och ofta kostnadseffektivt att inrätta en säkerhetszon för varje virtuell värddator, med mycket finmaskig filtrering som resultat.

Integration av filtreringen i virtualiseringsplattformen har dock baksidor, och den distribuerade trafikfiltreringen kan medföra utmaningar vid till exempel granskningar av säkerhetsskyddet. Det är till exempel inte alltid entydigt vilka det filtrerande systemets avgränsningar är, och de krav som i en traditionell miljö ställs på filtrerande enheter måste därför ofta utökas till att gälla hela virtualiseringsplattformen. Filtrering inom mjukvaradefinierade nätverk är därför också ofta ett *komplement* till den mer traditionella filtreringen som utförs av för ändamålet avsedda och separata system.

4.3 Redundans

Vid införande av redundanta systemlösningar för filtrering finns ett antal olika huvudfaktorer att ta hänsyn till. Trafik ska fördelas mellan de ingående elementen och eventuellt upprättade tillstånd behöver synkroniseras för att inte trafikströmmar ska brytas om någon komponent slutar fungera.

4.3.1 Vägval

Vägval mellan i systemlösningen ingående nätverkselement sker oftast genom att trafiken skickas mot en aktiv nod via en virtuell adress på datalänknivå, till exempel med hjälp av *Virtual Router Redundancy Protocol* (VRRP) [RFC5798], *Common Address Redundancy Protocol* (CARP) eller *Hot Standby Router Protocol* (HSRP) [RFC2281]. Det förekommer också att trafiken styrs med hjälp av dynamiska vägvalsprotokoll på nätverksnivå.

I de systemlösningar för redundans där flera noder är aktiva och hanterar trafik samtidigt skickas all trafik till samtliga noder, till exempel via *Ethernet multicast*, och endast den nod som avdelats för det aktuella kommunikationsflödet för detta vidare. Vilken nod som avdelas för vilka trafikflöden förhandlas inbördes mellan de ingående noderna. Alternativt fördelas trafiken av angränsande routrar med hjälp av ett dynamiskt vägvalsprotokoll med stöd för lastdelning via *Equal-Cost Multi-Path routing* (ECMP), till exempel *Border Gateway Protocol* (BGP) [RFC4271] eller *Open Shortest Path First* (OSPF) [RFC2328].

4.3.2 Synkronisering av kommunikationstillstånd

Då tillgänglighetskraven fordrar att kommunikationstillstånd ska kunna tas över av andra noder i en lösning som innefattar redundans måste de ingående noderna löpande utbyta information om upprättade tillstånd med varandra. Görs inte detta, kan paket komma att blockeras när trafik behöver hanteras av en annan nod än det som initialt hanterade kommunikationen vid upprättandet tillståndet.

Tillståndshanteringen blir mer komplex i de fall terminerande filtrering används. I generiska operativsystem går det ofta bra att synkronisera över tillståndet för den rena paketfiltrerande funktionen, men att synkronisera över kärnans anslutningar (*TCP/UDP sockets*) är desto svårare. Detta medför att det kan vara svårare att åstadkomma tillförlitlig redundans för applikationsprotokoll som bygger på långvariga sessioner och som inte sömlöst återupprättar kommunikationen vid kommunikationsfel, om filtreringslösningen baseras på generiska operativsystem.

4.4 Administration

Administrationsprinciperna för olika former av filtrerande system kan i huvudsak delas upp i två kategorier: lokal administration av ett enskilt system (möjligen bestående av flera redundanta noder), och central administration av flera filtreringssystem.

Lokal administration sker antingen direkt mot den filtrerande enheten via ett kommandoradsgränssnitt, *Command Line Interface* (CLI), eller via ett grafiskt gränssnitt. De grafiska gränssnitten kan antingen nyttja en tunn klientlösning, vanligen en webbläsare, eller en särskilt administrationsprogramvara som kommunicerar via en säker kontrollkanal.

Central administration implementeras vanligen i en dedicerad administrationsenhet skild från de filtrerande enheter som ska administreras. Ofta består administrationsenheten av flera ingående komponenter för till exempel administrationsgränssnitt, logghantering och övervakning.

För att minska de filtrerande enheternas angreppsyta har flera leverantörer valt att helt avskaffa det lokala administrationsgränssnittet, alternativt begränsa det lokala gränssnittet till att endast kunna konfigurera varifrån övrig konfiguration ska kunna ske.

Administrationsgränssnittet används normalt också för att underhålla enheternas egna programvara. Som för all säkerhetskritisk infrastruktur är en mycket viktig funktion att snabbt kunna distribuera uppdaterad programvara till alla, eller en delmängd av, de enheter som finns installerade.

4.4.1 Konfigurationsstyrning

Oavsett om administration sker lokalt eller centralt finns ett antal funktioner för konfigurationsstyrning som bör beaktas.

Atomisk konfiguration och versionshantering

För att den filtrerande enheten alltid ska ha ett väldefinierat konfigurationsläge är atomisk omkonfiguration önskvärt. Detta innebär att all omkonfiguration sker i transaktioner, det vill säga att samtliga ingående förändringar genomförs tillsammans eller inte alls. En vanligt förekommande funktion är automatiskt backa till ett tidigare känt fungerande konfigurationsläge om den filtrerande enhetens nya konfigurationen inte kan bekräftas av administratören efter aktivering, till exempel på grund av att enheten inte går att nå efter omkonfiguration.

För att enkelt kunna gå tillbaka till tidigare konfigurationslägen, och för att på ett enkelt sätt kunna identifiera vilka förändringar som skett vid en given omkonfiguration, bör all konfiguration versionshanteras.

Abstraktion

Definition av objekt och andra resurser som ingår i ett filtreringssystem bör kunna definieras inom hela den administrativa domänen. En samordnad objekt- och resurshantering möjliggör abstraktion av till exempel adresser och applikationsdefinitioner, med möjlighet till betydligt finmaskigare regelverk och mindre risk för felkonfiguration som resultat.

Programmatiska gränssnitt

När en filtreringsfunktion ska integreras med andra säkerhetssystem, och dessa dynamiskt ska kunna påverka den filtrerande enheten,

kan olika former av programmatiska gränssnitt, *Application Programming Interface* (API), användas för styrning av regelverket. Traditionellt har ofta *Simple Network Management Protocol* (SNMP) används för att styra nätverkselement, men på senare tid är *Network Configuration Protocol* (NETCONF) mer vanligt förekommande. Till skillnad från SNMP, som främst används för övervakning, är NETCONF från början konstruerat för att hantera konfigurationer av nätverkselement av den typ det är fråga om här.

4.4.2 Användarhantering och behörigheter

Administratörer som ges behörighet att administrera ett system för filtrering av kommunikation bör kunna definieras både lokalt och centralt. Beroende på hur systemet integreras kan central behörighetskontroll, till exempel via *Remote Authentication Dial-In User Service* (RADIUS) [RFC2865] eller *Lightweight Directory Access Protocol* (LDAP) [RFC1777], vara att föredra. För att inte hamna i ett låst läge om kommunikationen bryts bör det dock alltid säkerställas att det går att administrera den filtrerande enheten via någon nödmekanism.

Olika behörighetsnivåer i ett och samma system bör kunna användas för att begränsa åtkomst till läsa eller ändra konfiguration i systemets olika delar. Detta är framförallt viktigt i de fall den filtrerande enheten är uppdelad i flera logiska nätverkselement.

4.5 Loggning

En filtrerande enhet kan komma att producera stora mängder logginformation, bland annat innefattande:

Administrativ logg Information om administrativa åtgärder, till exempel när en administratör ansluter till den filtrerande enhetens administrationsgränssnitt, vilka åtgärder som vidtagits och vilka konfigurationsparametrar som ändrats och när.

Filterlogg Information om kommunikation som vidareförmedlats och blockerats genom den filtrerande enhetens regelverk. Loggning bör kunna konfigureras per regel samt om alla paket ska loggas eller bara första paketet i ett identifierat flöde.

Innehållslogg Applikationsnivåfilter kan i många fall konfigureras att logga applikationsdata som passerar den filtrerande enheten – antingen komplett data, eller metadata per transaktion.

Trafikdatalogg Information om trafikinnehåll som förmedlats via systemet. Detta kan vara sammanfattande trafikmetadata som överförs till ett separat analyssystem, eller kopior av utvalda eller samtliga paket.

Industristandard för att överföra generisk loggningformation är i praktiken SYSLOG [RFC5424], även om leverantörsspecifika protokoll ibland förekommer. Traditionellt transporteras SYSLOG över *User Datagram Protocol* (UDP), men kan även nyttja *Transmission Control Protocol* (TCP) och *Transport Layer Security* (TLS) [RFC5246]. Det senare är att föredra då det resulterar i bättre tillförlitlighet, transportskydd samt ursprungsidentifiering vid överföring av logginformation.

Överföring av trafikflödesdata sker vanligen med IPFIX [RFC7011] eller Netflow [RFC3954].

Förkortningar

ALG Application Layer Gateway

API Application Programming Interface

ARP Address Resolution Protocol

ASIC Application-Specific Integrated Circuit

BGP Border Gateway Protocol

CAM Content Addressable Memory

CARP Common Address Redundancy Protocol

CC Common Critieria

CLI Command Line Interface

DLP Data Loss Prevention

DNS Domain Name System

EAL Evaluation Assurance Level

ECMP Equal-Cost Multi-Path routing

FPGA Field-Programmable Gate Array

FTP File Transfer Protocol

GRE General Routing Encapsulation

HSRP Hot Standby Router Protocol

HTML Hypertext Markup Language

HTTP Hypertext Transfer Protocol

ICAP Internet Content Adaptation Protocol

IPS Intrution Prevention System

IPsec Internet Protocol Security

LDAP Lightweight Directory Access Protocol

MAC Media Access Control

NAC Network Admission Control

NDP Neighbor Discovery Protocol

NETCONF Network Configuration Protocol

NGFW Next Generation Firewall

OSPF Open Shortest Path First

PP Protection Profile

RADIUS Remote Authentication Dial-In User Service

SDN Software-defined Networking

SDP Session Description Protocol

SIP Session Initiation Protocol

SNMP Simple Network Management Protocol

SPI Stateful Packet Inspection

ST Security Target

TCP Transmission Control Protocol

TLS Transport Layer Security

UDP User Datagram Protocol

UTM Unified Threat Management

VLAN Virtual Local Area Network

VPN Virtual Private Network

VRF Virtual Routing and Forwarding

VRRP Virtual Router Redundancy Protocol

Sakregister

Referenser

[CC] ISO/IEC 15408-1:2005: Information technology – Security techniques – Evaluation criteria for IT security - Part 1: Introduction and general model.

[RFC1034] P.V. Mockapetris. Domain names - concepts and facilities. RFC 1034 (INTERNET STANDARD), november 1987. Updated by RFCs 1101, 1183, 1348, 1876, 1982, 2065, 2181, 2308, 2535, 4033, 4034, 4035, 4343, 4035, 4592, 5936.
http://www.ietf.org/rfc/rfc1034.txt.

[RFC1701] S. Hanks, T. Li, D. Farinacci och P. Traina. Generic Routing Encapsulation (GRE). RFC 1701 (Informational), oktober 1994.
http://www.ietf.org/rfc/rfc1701.txt.

[RFC1777] W. Yeong, T. Howes och S. Kille. Lightweight Directory Access Protocol. RFC 1777 (Historic), mars 1995. Obsoleted by RFC 3494.
http://www.ietf.org/rfc/rfc1777.txt.

[RFC2281] T. Li, B. Cole, P. Morton och D. Li. Cisco Hot Standby Router Protocol (HSRP). RFC 2281 (Informational), mars 1998.
http://www.ietf.org/rfc/rfc2281.txt.

[RFC2328] J. Moy. OSPF Version 2. RFC 2328 (INTERNET STANDARD), april 1998. Updated by RFCs 5709, 6549, 6845, 6860.
http://www.ietf.org/rfc/rfc2328.txt.

[RFC2865] C. Rigney, S. Willens, A. Rubens och W. Simpson. Remote Authentication Dial In User Service (RADIUS). RFC 2865 (Draft Standard), juni 2000. Updated by RFCs 2868, 3575, 5080, 6929.
http://www.ietf.org/rfc/rfc2865.txt.

[RFC3261] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley och E. Schooler. SIP: Session Initiation Protocol. RFC 3261 (Proposed Standard), juni 2002. Updated by RFCs 3265, 3853, 4320, 4916, 5393, 5621, 5626, 5630, 5922, 5954, 6026, 6141, 6665, 6878, 7462, 7463.
http://www.ietf.org/rfc/rfc3261.txt.

[RFC3507] J. Elson och A. Cerpa. Internet Content Adaptation Protocol (ICAP). RFC 3507 (Informational), april 2003.
http://www.ietf.org/rfc/rfc3507.txt.

[RFC3954] B. Claise. Cisco Systems NetFlow Services Export Version 9. RFC 3954 (Informational), oktober 2004.
http://www.ietf.org/rfc/rfc3954.txt.

[RFC4271] Y. Rekhter, T. Li och S. Hares. A Border Gateway Protocol 4 (BGP-4). RFC 4271 (Draft Standard), januari 2006. Updated by RFCs 6286, 6608, 6793.
http://www.ietf.org/rfc/rfc4271.txt.

[RFC4301] S. Kent och K. Seo. Security Architecture for the Internet Protocol. RFC 4301 (Proposed Standard), december 2005. Updated by RFC 6040.
http://www.ietf.org/rfc/rfc4301.txt.

[RFC5246] T. Dierks och E. Rescorla. The Transport Layer Security (TLS) Protocol Version 1.2. RFC 5246 (Proposed Standard), augusti 2008. Updated by RFCs 5746, 5878, 6176, 7465.
http://www.ietf.org/rfc/rfc5246.txt.

[RFC5424] R. Gerhards. The Syslog Protocol. RFC 5424 (Proposed Standard), mars 2009.
http://www.ietf.org/rfc/rfc5424.txt.

[RFC5798] S. Nadas. Virtual Router Redundancy Protocol (VRRP) Version 3 for IPv4 and IPv6. RFC 5798 (Proposed Standard), mars 2010.
http://www.ietf.org/rfc/rfc5798.txt.

[RFC6698] P. Hoffman och J. Schlyter. The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA. RFC 6698 (Proposed Standard), augusti 2012.
http://www.ietf.org/rfc/rfc6698.txt.

[RFC7011] B. Claise, B. Trammell och P. Aitken. Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information. RFC 7011 (INTERNET STANDARD), september 2013.
http://www.ietf.org/rfc/rfc7011.txt.

[RFC7230] R. Fielding och J. Reschke. Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing. RFC 7230 (Proposed Standard), juni 2014.
http://www.ietf.org/rfc/rfc7230.txt.

[RFC793] J. Postel. TRANSMISSION CONTROL PROTOCOL. RFC 793 Standard, September 1981.
http://www.ietf.org/rfc/rfc793.txt.

[RFC959] J. Postel och J. Reynolds. File Transfer Protocol. RFC 959 (INTERNET STANDARD), oktober 1985. Updated by RFCs 2228, 2640, 2773, 3659, 5797, 7151.
http://www.ietf.org/rfc/rfc959.txt.