**Autentificarea in rețele**

**Rețele de calculatoare**

**Temă de casă**

**Profesor coordonator: Studenti:**

**Conf. dr. ing. Stefan Stăncescu** Matragociu Bogdan

Mîrzac Adrian

442A

**Cuprins**

Matragociu Bogdan:

1. Arhitectura AAA

2. Protocolul RADIUS

3. Framework-ul EAP

4. Protocolul PPPoE

Mirzac Adrian:

5. Protocolul PPPoA

6. Protocolul LDAP

7. Protocolul HDLC

8. Protocolul NCP

9. Bibliografie

1. **Arhitectura AAA (**Authentication, Authorization, Accounting **)**

Conceptul de AAA (Authentication, Authorization, Accounting) a apărut pentru simplificarea dezvoltării arhitecturilor de securitate în contextul creșterii complexității rețelelor. Decuplarea funcționalităților permite folosirea doar acelor componente ce sunt relevante pentru soluționarea nevoilor de securitate ale organizației.

Autentificarea (authentication) se referă la validarea identității utilizatorului ce dorește să folosească un serviciu securizat. Utilizatorul poate fi o entitate umană ce interacționează prin intermediul unui client, sau un program automat precum un protocol dinamic de ruta re. Exemplele de servicii securizate ce pot fi protejate includ accesul la consola unui echipament, accesul la rețeaua internă a unei companii sau la resursele unui server. Autentificarea se ocupă strict de validarea identității, nu și de privilegiile ce sunt ulterior oferite respectivei identități. Pentru acest aspect al securității se folosește componenta de autorizare.

Autorizarea (authorization) dictează ce acțiuni poate efectua o identitate autentificată. De exemplu, unui utilizator ce accesează consola unui echipament de rețea i se poate permite doar afișarea configurațiilor, nu și modificarea acestora. La introducerea unei comenzi, componenta de autorizare din arhitectura de securitate AAA trebuie să efectueze consultarea privilegiilor utilizatorului, după care se returnează un răspuns de autorizare cu succes sau autorizare eșuată.

Jurnalizarea (accounting) realizează o evidentă a acțiunilor efectuate de identitatea autentificată. Evenimentele relevante ce pot fi jurnalizate includ, de exemplu, inițializarea unei sesiuni de terminal sau terminarea respectivei sesiuni. Prin intermediul unei jurnalizări adecvate se poate examina ulterior, în cazul apariției unei probleme, ce comenzi au fost efectuate și de către cine. Rezultatul este un proces de depanare simplificat și responsabilizarea utilizatorilor implicați.

Cele trei aspecte ale conceptului de AAA pot fi folosite în funcție de cerințele rețelei. O rețea simplă cu un singur administrator poate avea nevoie doar de componenta de autentificare, urmând ca în urma autentificării să îi fie oferite utilizatorului toate privilegiile. O rețea complexă cu administratori multipli poate necesita componenta de autorizare pentru a gestiona roluri multiple (de exemplu, un administrator de sistem poate necesita mai multe drepturi decât un utilizator responsabil de soluționarea tichetelor).

Sumarizând, autentificarea răspunde la întrebarea "Cine ești?", autorizarea la "Ce poți să faci?" și jurnalizarea la "Ce ai făcut?".

Pentru a putea fi folosită arhitectura AAA, echipamentele de rețea în general oferă suport pentru protocoale care implementează funcționalitățile de autentificare, autorizare și jurnalizare. Exemple de astfel de protocoale includ RADIUS, TACACS+ și Diameter. Pe lângă acestea, funcționalitățile AAA pot fi implementate și prin alte soluții, precum folosirea unor utilizatori configurați local pentru autentificare sau folosirea de log-uri locale pentru jurnalizare.

1. Protocolul RADIUS ( Remote Access Dial-ln User Service )

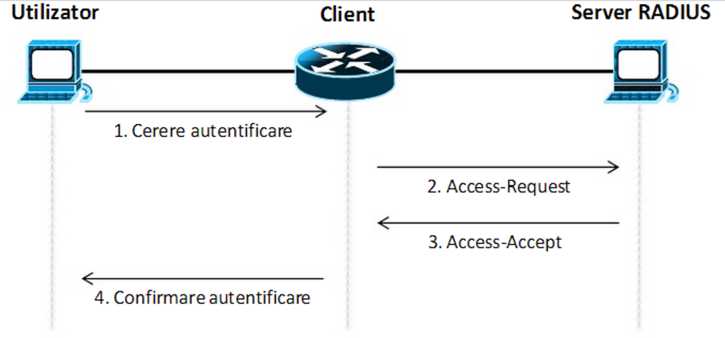
RADIUS (Remote Access Dial-ln User Service) este un protocol deschis ce definește cum trebuie realizată comunicația între un echipament de rețea ce necesită funcționalități AAA și un server central ce furnizează aceste funcționalități. Protocolul RADIUS este independent de resursa protejată, ceea ce îl face compatibil cu un număr ridicat de scenarii de utilizare. De exemplu, un ruter poate folosi un server RADIUS pentru a gestiona utilizatorii ce au acces la configurație și pentru a jurnaliza evenimentele de acces. Avantajul RADIUS este că la același server se pot conecta mai mulți clienți, multiple echipamente eterogene (rutere, switch-uri, server Linux, staţii de lucru) putând accesa astfel aceeaşi bază de date de drepturi şi utilizatori.

RADIUS a fost introdus în în 1991, şi a fost definit ulterior în RFC 2058 [1], publicat în 1997. Protocolul a suferit numeroase modificări de la versiunea iniţială, versiunea actuală fiind definită în RFC 2865 [2]. Pe lângă acesta, există multiple alte RFC-uri ce definesc funcţionalităţi specifice sau extensii ale protocolului. Dezvoltarea RADIUS continuă şi în ziua de azi; de exemplu, în anul 2012 protocolul RADIUS a fost extins pentru a permite şi funcţionarea peste TCP, în loc de strict UDP conform specificării iniţiale. Una din limitările importante ale protocolului este lipsa de separare a conceptelor de autentificare şi autorizare din arhitectura AAA. Cu alte cuvinte, protocolul RADIUS nu poate fi folosit pentru a determina dacă un utilizator are sau nu dreptul de a efectua o anumită comandă. Această limitare nu este prezentă şi în Diameter, protocol proiectat ca succesor al RADIUS. Adoptarea Diameter este până în prezent redusă, în mare parte deoarece numeroase echipamente de reţea nu oferă încă suport pentru acesta.

RADIUS funcţionează peste UDP sau TCP, implicit fiind folosit UDP cu portul 1812 pentru autentificare/autorizare şi 1813 pentru jurnalizare.

In terminologia RADIUS, utilizatorul este entitatea care doreşte să se autentifice, clientul este entitatea pe care utilizatorul încearcă să se autentifice şi serverul este echipamentul care validează autentificarea prin intermediul protocolului RADIUS.

De exemplu, dacă Alice este un administrator de reţea ce doreşte să configureze un ruter, atunci ea reprezintă utilizatorul. Ruterul ce trebuie configurat este în acest context clientul, iar serverul RADIUS poate fi un server Linux din reţea. în general, staţia utilizatorului nu trebuie să vorbească protocolul RADIUS pentru a se putea autentifica. Acesta este folosit strict pentru comunicaţia dintre client şi server. Există situaţii în care se doreşte un procedeu mai complex de autentificare (de exemplu, pe bază de token); în astfel de situaţii poate fi necesară stabilirea unui tunel RADIUS direct între utilizator şi server.



Autentificare cu succes folosind un server RADIUS

Protocolul RADIUS foloseşte multiple formate de pachete, diferenţierea făcându-se prin intermediul câmpului de un octet Code, situat la începutul pachetului. Codurile existente sunt:

* 1: Access-Request
* 2: Access-Accept
* 3: Access-Reject
* 4: Accounting-Request
* 5: Accounting-Response
* 11: Access-Challenge
* 12: Status-Server
* 13: Status-Client
* 255: Rezervat

In cadrul acestor formate de mesaje pot exista mai multe atribute, specificate prin structuri TLV (Type/Length/Value). Atributele au rolul de a preciza informaţiile relevante mesajului, precum adresa IP, numele utilizatorului sau starea sesiunii de jurnalizare. în continuare sunt detaliate două dintre scenariile de bază pentru funcţionarea protocolului RADIUS.

**Autentificare**

La încercarea de conectare din partea utilizatorului, ruterul trebuie să valideze datele de conectare (care pot fi user/parolă, cheie privată, token, etc.). Primul pas în realizarea acestei validări este trimiterea unui mesaj Access-Request către serverul RADIUS. în interiorul mesajului sunt incluse datele de acces furnizate de către utilizator. Serverul RADIUS răspunde apoi printr-unul din trei mesaje:

* Access-Accept, în cazul în care datele furnizate sunt valide
* Access-Reject, în cazul în care autentificarea a eşuat
* Access-Challenge, dacă finalizarea autentificării necesită informaţii suplimentare din partea utilizatorului

**Jurnalizare**

Procesul de realizare a jurnalizării RADIUS este definit în RFC 2866 [3]. Odată realizată autentificarea, clientul poate deschide o sesiune de jurnalizare prin trimiterea unui mesaj Accounting-Request, cu atributul Acct-Status-Type pus pe valoarea Start. Dacă jurnalizarea s-a făcut cu succes, serverul trimite înapoi un mesaj Accounting-Response. Dacă jurnalizarea a eşuat, atunci nu este trimis niciun răspuns. După deschiderea sesiunii, clientul trimite periodic actualizări folosind acelaşi format de mesaj Accounting-Request, cu Acct-Status-Type pus pe valoare Interim-Update. Pe lângă aceste atribute, mesajul poate conţine şi informaţii suplimentare despre evenimentul jurnalizat. Finalizarea sesiunii se face cu un mesaj Accounting-Request, cu Acct-Status-Type pe Stop.

1. Protocolul PPPoE (Point to Point Protocol over Ethernet )

PPP (Point to Point Protocol) este un protocol de nivel 2 ce este în general folosit de către furnizorii de servicii pentru a oferi clienţilor acces Internet. Deşi este situat la acelaşi nivel din stiva OSI ca şi protocolul Ethernet, PPP oferă funcţionalităţi suplimentare ce îl fac atractiv pentru folosirea în furnizarea de servicii. Aceste funcţionalităţi includ: compresia traficului, autentificarea utilizatorului şi criptare.

Spre deosebire de Ethernet, care este un protocol multiacces, PPP este un protocol punct la punct. Acesta poate fi deci folosit pentru a transmite trafic IP peste medii fizice punct la punct precum cablu serial sau linii telefonice.

Uneori, reţelele furnizorilor de servicii folosesc Ethernet pentru a transporta traficul IP dintre clienţi şi reţeaua furnizorului. De exemplu, este mai ieftin şi mai simplu de instalat o infrastructură Ethernet într-un complex studenţesc în loc de folosirea altor medii de transport. Dezavantajul Ethernet este absenţa autentificării, ceea ce duce la dificultăţi în gestionarea unor politici avansate pe bază de identitatea utilizatorului ce se conectează la Internet.

Abordarea cea mai simplă ar fi folosirea PPP ca protocol de nivel 2, însă acesta este un protocol punct la punct şi nu poate funcţiona direct peste reţelele fizice multiacces folosite de Ethernet. Soluţia este rularea PPP peste Ethernet, prin intermediul unui protocol numit PPPoE (PPP over Ethernet).

PPPoE a fost introdus în RFC 2516 [5], ca o soluţie eficientă din punct de vedere al costurilor pentru furnizorii de servicii. Aceştia foloseau deja PPP pentru autentificare pe linii mai vechi (dial-up), însă schimbarea mediului de transmisie către cablu torsadat atât pe segmentul ISP-Client cât şi în reţeaua clientului a dus la incompatibilităţi cu sistemele existente. Protocolul este implementat ca un nivel intermediar între Ethernet, care stă direct peste infrastructura fizică folosită, şi PPP, care este necesar pentru a oferi autentificare şi compatibilitate cu infrastructura furnizorului de servicii.

|  |  |
| --- | --- |
| Nivel reţea | IP |
| Nivel legătură de date | PPP |
| PPPoE |
| Ethernet |
| Nivel fizic | Cablu torsadat / Fibră optică |

Stiva de protocoale în cazul folosirii PPPoE

1. Framework-ul EAP ( Extensible Authentication Protocol )

Multitudinea metodelor de autentificare a dus la nevoia creării unor standarde ce permit integrarea unui set larg de metode de autentificare cu infrastructurile care le utilizează. De exemplu, un server AAA care implementează un astfel de standard poate utiliza orice metodă compatibilă fără a fi necesară adăugarea de cod direct în server. Integrarea cu metoda EAP se face prin module ce expun serverului o interfaţă comună de operaţii.

EAP (Extensible Authentication Protocol) este un framework c e defineşte formatul mesajelor de reţea pentru o multitudine de mecanisme de autentificare. EAP este folosit de către numeroase protocoale de securitate. De exemplu, RADIUS şi WPA2 folosesc metode EAP pentru autentificare.

Terminologia folosită de EAP este aceeaşi cu cea folosită de către standardul 802.1X. Acesta defineşte următoarele dizpozitive:

* Autentificatorul (Authenticator) - echipamentul care iniţiază comunicaţia EAP;
* Solicitatorul (Supplicant) - echipamentul care răspunde autentificatorului;
* Serverul de autentificare (Backend authentication server / AAA Server) - echipamentul care oferă autentificatorului serviciul de autentificare.

Aceste dispozitive sunt exact aceleaşi din scenariul cu RADIUS prezentat anterior în acest capitol. Utilizatorul din scenariul respectiv este un exemplu de solicitator, clientul (echipamentul de reţea, de exemplu, un ruter) este autentificatorul, iar serverul de autentificare poate fi un server RADIUS.

Comunicaţia EAP este formată din patru paşi. în primul pas, autentificatorul cere solicitatorului să înceapă autentificarea printr-un mesaj tip Request. în al doilea pas, solicitatorul răspunde autentificatorului cu un mesaj Response oferind datele cerute de acesta. Ambele mesaje conţin un câmp de tip (Type) care spune ce metodă EAP este folosită. în al treilea pas, autentificatorul poate trimite un nou Request pentru alte informaţii către solicitator, iar acesta trebuie să răspundă cu informaţii valide. Acest schimb se poate repeta de oricâte ori, până când autentificatorul deţine toate informaţiile necesare pentru a completa procesul de autentificare. Ultimul pas este reprezentat de trimiterea unui mesaj de tip EAP Success sau EAP Failure către solicitator, în funcţie de rezultatul autentificării.

In unele cazuri, autentificatorul poate să nu aibă implementată local metoda EAP şi să se comporte ca un proxy pentru serverul de autentificare din spate. Această funcţionalitate permite unor echipamente de reţea simple precum un switch sau un access point să folosească metode EAP complexe fără a fi necesar cod suplimentar.

Unul din avantajele oferite de EAP este flexibilitatea acestuia, întrucât metoda de autentificare poate fi recunoscută direct din mesajul primit, nefiind nevoie de configurări explicite pe serverul de autentificare.

RFC-ul EAP defineşte un set de mesaje de bază. Dintre acestea, mesajele de tip 1 până la 4 trebuie suportate de toate implementările EAP.

Identity (tip 1). Un mesaj Request de tip Identity este folosit pentru a descoperi informaţii despre identitatea solicitatorului şi pentru a descoperi ce metode EAP urmează să fie folosite. Un exemplu de astfel de mesaj poate să ceară numele de utilizator ce urmează să fie folosit pentru a configura echipamentul, însă utilizarea în acest fel a tipului Identity este descurajată deoarece mesajele sunt transmise necriptat şi pot fi citite sau modificate de un atacator.

Notification (tip 2). Acest tip de mesaj este folosit pentru a informa utilizatorul despre un anumit eveniment, de exemplu expirarea perioadei de viaţă a unei chei de unică folosinţă. în general nu este necesară folosirea acestui tip.

Nak (tip 3). Mesajele de tip Nak sunt doar de tip Response şi sunt folosite pentru a sugera autentificatorului că o anumită metodă de autentificare este inacceptabilă. Un mesaj de tip Nak ar trebui să conţină şi metodele de autentificare pe care solicitatorul le consideră valide.

MD5-Challenge (tip 4). Acest tip de mesaj este folosit de către autentificator pentru a verifica existenţa unei chei specifice la solicitator. Un mesaj de tip Request ce conţine un şir aleator este trimis solicitatorului, care trebuie să răspundă printr-un mesaj de tip Response ce conţine un hash al şirului aleator. Calculul hash-ului trebuie să aibă ca intrare şi o cheie secretă. Acest tip de autentificare permite verificarea cheii fără ca aceasta să fie transmisă peste reţea.

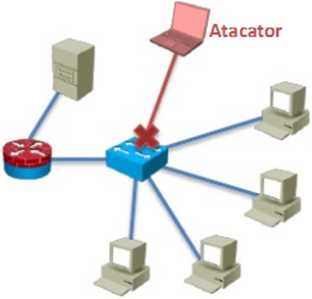
One-Time Password - OTP (tip 5). Acest tip de mesaj are o funcţionalitate similară cu MD5- Challenge, cu diferenţa că în calcularea hash-ului nu se foloseşte o cheie secretă permanentă anterior stabilită, ci o cheie de unică folosinţă.

Generic Token Card - GTC (tip 6). Acest tip de mesaj este proiectat pentru a fi folosit în combinaţie cu dispozitive tip Token care generează un cod de autentificare valabil pentru o perioadă limitată de timp.

Datorită flexibilităţii pe care o oferă, metodele EAP sunt folosite în numeroase standarde moderne de securitate. Câteva exemple sunt prezentate în continuare: 802.IX pentru reţele cu fir şi WPA Enterprise pentru reţele wireless.

* 1. 802.IX

Funcţionarea EAP peste reţele de tip IEEE 802.3 (de exemplu. Fast Ethernet) este definită în cadrul standardului 802.IX. Standardul defineşte cum se încapsulează informaţia EAP în cadrul frame-ului IEEE802.3, încapsulare ce poartă numele de EAPOL (EAP over LAN).



Prevenirea atacurilor prin implementarea 802.1x

802. IX este folosit în reţele unde se doreşte autentificarea utilizatorilor la nivel 2. De exemplu, un switch IEEE802.3 poate să solicite din partea unui echipament nou conectat un user şi o parolă înainte să permită accesul la reţeaua din spate. Un server AAA compatibil EAP (de exemplu, RADIUS) este consultat de switch în vederea validării informaţiilor de autentificare oferite de dispozitiv. Avantajul principal al autentificării de nivel 2 este că un posibil atacator nu poate obţine deloc acces la reţea până nu se autentifică, limitându-se astfel accesul acestuia la serviciile complexe din reţea.

* 1. WPA Enterprise

Standardul de securitate pentru reţele wireless 802.11 WPA/WPA2 (WiFi Protected Access) poate fi utilizat în două variante:

* PSK (pre-shared key), când se doreşte autentificarea directă a utilizatorilor de către echipament
* Enterprise, când autentificarea se face prin intermediul unui server RADIUS

A doua variantă mai este cunoscută şi sub numele WPA-802.1X şi este definită în cadrul standardului 802.IX. Autentificarea se face folosind metode EAP, permiţându-se astfel modele de autentificare mai complexe decât simpla autentificare prin cheie partajată permisă în WPA-PSK. Pentru a putea certifica un echipament wireless ca fiind compatibil WPA/WPA2 trebuie implementat un set minim de metode EAP, definite de către organizaţia de standardizare Wi-Fi Alliance. Exemple de astfel de metode includ în 2013: EAP-TLS, PEAP-TLS, EAP-SIM, EAP-AKA şi EAP-FAST. Acestea trebuie să fie implementate atât de clientul wireless care încearcă să se autentifice în reţea, cât şi de serverul RADIUS utilizat pentru autentificare.

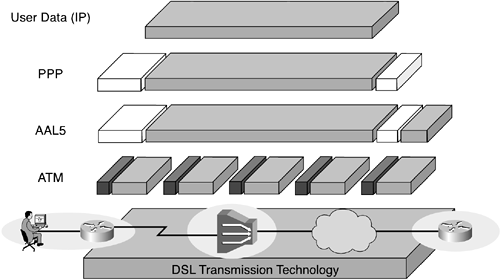
1. **Protocolul PPPoA (PPP over ATM - Asynchronous Transfer Mode)**

PPP este definit pentru utilizarea in special cu RFCs 1331 si 1661 ca o metodă standard de încapsulare a diferitelor tipuri de protocoale de nivel superior peste conexiuni point-to-point. Extinde structura datelor la nivel înalt structura pachetului HDLC cu un identificator de protocol de 16 biţi care conţine informaţii despre conţinutul pachetului.

PPP este definit pentru utilizarea în mod special cu AAL5 (ATM Adaptation Layer 5) în RFC 2364. PPPoA operează în oricare două moduri — SRL/SNAP sau Virtual Circuit-multiplexing (VC-mux).

În modelul PPPoA, aşa cum se arată în figura 3-9, ATU-R încapsulează datele utilizatorului (PDUs) in diferite moduri în cadrele de PPP. Cu PPPoA, sesiunea de PPP este între ATU-R şi agregarea ruterului.

Link-ul de PPP este reziliat la agregarea furnizorului ruterului.



Protocoalele PPPoA

Pachetele PPP includ două protocoale vitale, precum şi de trafic de utilizator-LCP şi unul dintre mai multe tipuri de NCPs. LCP stabileşte, configurează, testeaza, menţine şi termina conexiunea de date pentru a se asigura de calitatea liniei ca este pentru transmiterea protocoalelor L3. Un al doilea protocol folosit de PPP este unul din multele tipuri de NCPs. Acest lucru vine în funcțiune de link-ul de date stabilit de LCP. NCP stabileşte şi configurează protocoale aplicabile de retea, pentru că PPP este proiectat sa permita utilizarea simultană a mai multor protocoale de retea. NCP contine informatii despre protocoale din stratul superior, inclusiv IP şi IPX. De exemplu, NCP PPP pentru IP este IPCP.

Stiva protocolului PPPoA:

Stiva generala a protocolului PPPoA este afisata mai jos in figura 3-10.

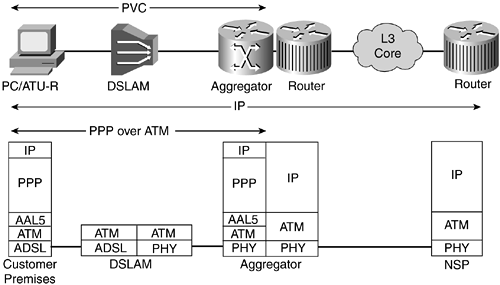


Fig. 3-10

Pornind de la punctul de vedere al utilizatorului, PC-ul transmite un cadru Ethernet, care poartă un pachet IP. ATU-R primeste un cadru Ethernet, elimină antetul Ethernetului antet şi adaugă un antet de PPP. În afară de datele utilizatorului, ATU-R include informații specifice PPP, cum ar fi parametri de securitate pentru transportul peste bucla ADSL şi DSLAM.

Avantajele si dezavantajele PPPoA:

Avantajele sunt urmatoarele:

* Arhitectura moşteneşte cele mai multe dintre avantajele PPP utilizate în modelul de apelare.
* O implementare PPPoA implică configurarea ATU-R cu informații de autentificare PPP (login si parola). Acest lucru este un avantaj major al PPPoA peste simple implementări IRB şi RBE, care nu au nicio garanție. Asta este, PPP permite per sesiune AAA. Per sesiunea AAA permite furnizorului de servicii sa încărce abonatul bazat pe sesiune timp pentru diferite servicii oferite.
* Conservarea adresei IP la modemul DSL permite furnizorului de servicii sa atribuie numai o adresa de IP pe abonament. Când ATU-R este configurat pentru NAT, toţi utilizatorii în spatele ATU-R pot utiliza o singură adresă IP pentru a ajunge la destinaţii diferite.
* Gestionarea IP pentru NAP/NSP la fiecare utilizator individual este redusă în timp ce conservarea adresei IP.
* NAPs/NSPs ofera acces sigur la gateway-uri corporative, fără PVC end-to-end, folosind un circuit Layer 3 sau L2F/L2TP tuneluri de gestionare. Prin urmare, furnizorii de servicii pot scala propriile modele de afaceri pentru vanzare cu servicii ridicate.
* Rezolvarea problemelor individuale ale abonaţilor este un alt avantaj al PPPoA. NSP poate identifica cu uşurinţă care abonaţi sunt on sau off in sesiunile active PPP, mai bine decât cautarea in intreg grupurile aşa cum este cazul cu reducerea arhitecturii.
* NSP poate suprascrie prin desfăşurarea inactiva şi sesiune timeout utilizând funcţia Radius pentru fiecare abonat.

Dezavantajele PPPoA sunt urmatoarele:

* Complexitatea relativă de rutare de configurare pe mai multe ATU-R pentru PPPoA creşte costurile forţei de muncă si hardware.
* PPPoA are numai o singură sesiune pe VC. Deoarece numele de utilizator şi parola sunt configurate pe UTA-R, toţi utilizatorii în spatele modemul DSL pot utiliza numai un singur set de servicii. Utilizatorii nu pot selecta aleatoriu seturi diferite de servicii, dar pot folosi VCs şi pot stabili diferite sesiuni de PPP pe diferite VCs. Desigur, folosind mai multe VCs creşte complexitatea de configurare, în continuare creşterea costurilor forţei de muncă şi necesitatea de a instrui personalul de asistenţă serviciu .
* Prestatorul de servicii trebuie să menţină o bază de date, numele de utilizator şi parole pentru toţi abonaţii. Dacă se folosesc tuneluri sau servicii de proxy, autentificarea se poate face pe baza numelui si portalului corporatist. Acest lucru reduce dimensiunea bazei de date pe care prestatorul de servicii trebuie să o menţină.

Rezumat PPPoA:

* PPPoA (RFC 2364) foloseste AAL5 ca protocol şi sprijină toate opţiunile de cartografiere ATM.
* PPPoA este arhitectura de alegere pentru mulţi furnizori de servicii, deoarece este foarte scalabil, pentru furnizorii şi utilizatorii finali, şi permite functionalitatea L3, inclusiv securitate.
* Consideraţiile pentru implementarea PPPoA includ costuri suplimentare ale functionalitatii mai mari.

1. **Protocolul LDAP ( Lightweight Directory Access Protocol )**

Scopul:

Lightweight Director Access Protocol (LDAP) este un protocol de serviciu, care ruleaza pe un strat deasupra stivei TCP/IP. Acesta prevede un mecanism utilizat pentru a conecta, a conecta, şi modifca directoarele Internet.

Serviciul director LDAP este bazat pe un model de client-server. Funcţia de LDAP este de a permite accesul la un director existent.

Modelul de date (date si namespace) de LDAP este similar cu cel al serviciului de Director X.500 OSI, dar cu necesarul de jos.

Asocierile LDAP API simplifică scrierea in directoriul Internet a cererilor de serviciu.

Unde este aplicat:

LDAP API se aplică la directoriul management şi browser-ul aplicaţii care nu au serviciul suport ca funcţia lor primara. Dimpotrivă, LDAP nu este nici pentru crearea de directoare, nici nu specifică modul în care funcţionează un serviciu de Director.

Audienta dezvoltarii:

Documentatia LDAP API în platforma Software Development Kit (SDK) este destinata pentru experimentarea in limbajele de programare C si C++ şi dezvoltatorii de Director Internet. LDAP sprijină limbajele de programare C şi C++ .O familiaritate cu serviciile de Director şi modelul Client/Server LDAP sunt necesare pentru dezvoltarea cu LDAP API.

Cerintele de rulare:

Aplicaţiile client care utilizează LDAP API, ruleaza pe Windows Vista şi Windows XP. Toate platformele trebuie sa aibaTCP/IP instalat.

Serverele Active Directory, care suportă aplicaţiile client folosind LDAP API includ Windows Server 2008 şi Windows Server 2003.

1. **Protocolul HDLC ( High-level data link control )**

Se examineaza un grup de protocoale strans legate, putin mai vechi dar care sunt inca foarte utilizate. Ele sunt toate derivate din protocolul pentru legatura de date utilizat in lumea mainframe-urilor IBM, numit SDLC(Synchronimus Data Link Control). Dup ace a dezvoltat SDLC, IBM l-a supus examinarii ANSI si ISO pentru acceptare ca standard SUA si respective international.ANSI a modificat protocolul astfel inca acesta a devenit ADCCP ( Advanche Data Communication Control Procedure), iar ISO a modificat si a produs HDLC( High-Level Data Link Control).

Aceste protocoale se bazeaza pe aceleasi principia. Toate sunt orientate pe biti si folosesc inserarea de biti pentru transparenta datelor. Ele difera doar in puncte minore si totusi suparatoare. Toate protocoalele orientate pe biti folosesc structura de cadru prezentata in fig. 2. Campul adresa este primul ca importanta pentru liniile cu terminale multiple unde el este folosit pentru a identifica unul dintre terminale. Pentru liniile punct la punct el este folosit uneori pentru a desebi comenzile de raspunsuri.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 01111110 | Adresa | Control | date | Suna de control | `01111110 |

8 8 8 >0 16 8 biti

Fig 2.

Campul control este folosit pentru numere de secventa confirmari si alte scopuri dupa cum se va arata in continuare.

Campul date poate contine informatii arbitrare. Poate avea lungime arbitrara, cutoate ca eficienta sumei de control scade odata cu cresterea lungimii cadrului, datorita cresterii probabilitatii de aparitie a erorilor in rafala.

Campul Suma de Control este o variant CRC folosind tehnica prezentata in Sec. 3-2.2.

Cadrul este delimitat cu o alta secventa indicator . Pe liniile punct-la-punct inactive secventele indicator sunt transmise continuu. Un cadru minim contine trei campuri si are in total 32 de biti, excluzand indicatorii de capete.

Exista trei tipuri de cadre: Informatie, Supervizor si Nenumerotat. Continutul campului Control pentru fiecare dintre aceste trei tipuri este prezentat in fig. 3. Acest protocol foloseste o fereastra glisanta cu un numar de secventa reprezentat pe 3 biti. In fereastra pot fi pastrate, la un moment dat, pana la sapte cadre neconfirmate. Campul Secventa din fig. 3 este numarul de secventa al cadrului. Campul Urmator este o confirmare atasata. Oricum, toate protocoalele adera la conventia ca, in loc sa ataseze numarul ultimului cadru receptionat corect, sa foloseasca numarul primului cadru nereceptionat( adica urmatorul cadru asteptat), Optiunea pentru ultimul cadru primit sau urmatorul cadru receptionat este arbitrara; nu are importanta ce conventie este utilizata , daca este folosita cu consecventa.

|  |  |  |  |
| --- | --- | --- | --- |
| 0 | Secventa | P/F | Urmator |

a) biti 1 3 1 3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 0 | Tip | P/F | Urmator |

b)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 1 | Tip | P/F | Modificator |

c)

fig.3

Campul Control pentru a) un cadru de informative, b) un cadru de supervizare, c) un cadru nenumerotat.

Bitul P/F inseamna Test/ Final . El este folosit atunci cand un calculator (sau un concentrator) interogheaza un grup de terminale. Cand este folosit ca P, calculatorul invita terminalul sa trimita date. Toate cadrele trimise de terminal, cu exceptia celui final, au bitul P/F setat pe P. Pentru cadrul final vitul este setat la F.

Nivelul Legaturii de date in Internet

Internetul consta in masini individuale si o infrastructura de comunicatie care le conecteaza. In cadrul unei singure cladiri sunt larg utilizate LAN-urile pentru interconectare, dar infrastructura de arie larga este construita din linii inchiriate, punct la punct.

In practica comunicatie punct-la-punct este folosita in principal in doua situatii. In primul rand, mii de organizatii au una sau mai multe LAN-uri, fiecare cu un anumit numar de calculatoare gazda, statii de lucru ale utilizatorilor, servere si un ruter. Adesea ruterele sunt interconectate printr-un trunchi LAN. In mod tipic, toate conexiunile cu minea exterioara se fac printr-unul sau doua rutere care au lunii punct-la-punct inchiriate spre rutere aflate la distanta. Internet-ul este contruit din aceste rutere si liniile lor inchiriate care realizeaza subretelele de comunicatie.

A doua situatie in care liniile punct-la-punct joaca un rol major in internet o reprezinta milioanele de utilizatori individuali care au conexiuni de acasa la Internet folosind modemuri si linii telefonice commutate. De obicei PC-ul de acasa al unui utilizator apeleaza ruterul unui furnizor de servicii Internet si poate actiona astfel ca orice calculator gazda Internet. Aceasta metoda de operare nu este diferita de aceea in care exista o linie inchiriata intre PC si ruter, cu exceptia faptului ca atunci cand utilizatorul isi termina sesiunea, conexiunea va fi inchisa.

1. **PPP cu NCP (Network Control Protocols )**

Odată ce link-ul și PPP parametrii au fost negociate cu LCP, nodurile PPP apoi folosesc o serie de protocoale de control de rețea (NCP) pentru a negocia parametrii de protocoale individuale LAN. Microsoft PPP suporta următoarele NCP:

* Internet Protocol Control Protocol (IPCP) să negocieze utilizarea de IP.
* Internetwork Packet Exchange Control Protocol (IPXCP) să negocieze utilizarea IPX.
* AppleTalk Control Protocol (ATCP) să negocieze utilizarea AppleTalk.
* NetBIOS Rame Control Protocol (NBFCP) să negocieze utilizarea NetBEUI.

**IPCP**

Internet Protocol Control Protocol (IPCP), așa cum este folosit de către peer-urile Microsoft PPP este documentat în RFC 1332 și 1877. IPCP negociază parametri bazati pe IP pentru a configura dinamic un peer PPP bazate pe IP TCP într-o legătură punct-la-punct. Opțiuni comune IPCP includ o adresă IP și adresele IP de DNS și server de nume NetBIOS.

IPCP utilizează PPP Protocol ID-ul de 0x80-21. Structura pachet de IPCP este exact la fel pentru LCP, cu excepția ca numai tipurile de pachete de la 1 la 7 sunt definite. Pentru pachetele Configure-Request, Configure-Ack, Configure-Nack, and Configure-Reject , porțiunea de date a pachetului IPCP constă din una sau mai multe opțiuni IPCP. Fiecare opțiune IPCP constă dintr-un câmp de tip Option Type Field, Option Lenght care include lungimea totala in biti a optiunii, si datele asociate optiounii.

**Optiuni IPCP**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nume optiune** | **Tip optiune** | **Lungime Optiune** | **Descriere** |
| Protocol de compresie al IP-ului | 2 | 4 | Protocolul Van Jacobsen de compresie TCP. |
| Adresa IP | 3 | 6 | Adresa IP ce va fi alocata clientului remote. |
| Adresa server DNS primara | 129 sau 0x81 | 6 | Adresa primara server DNS a clientului. |
| Adresa server NBNS primara | 130 sau 0x82 | 6 | Adresa primara server NBNS (WINS) a clientului. |
| Adresa server DNS secundara | 131 sau 0x83 | 6 | Adresa secundara server DNS a clientului. |
| Adresa server NBNS secundara | 132 sau 0x84 | 6 | Adresa secundara server NBNS (WINS) a clientului. |

### IPXCP

### Internetwork Packet Exchange Control Protocol (IPXCP), așa cum este folosit de către peer-urile Microsoft PPP este documentat în RFC, 1552. IPXCP negociază parametri bazati pe IPX pentru a configura dinamic peer-urile PPP IPX într-o legătură punct-la-punct. Opțiuni comune IPXCP includ rețea IPX și adrese de nod.

IPXCP utilizează PPP Protocol ID-ul de  0x80-2B. Structura pachet de IPXCP este exact la fel cu LCP, cu excepția ca numai tipurile de pachete de la 1 la 7 sunt definite. Pentru pachetele Configure-Request, Configure-Ack, Configure-Nack, and Configure-Reject , porțiunea de date a pachetului IPXCP constă din una sau mai multe opțiuni IPXCP . Fiecare opțiune IPXCP constă dintr-un câmp de tip Option Type Field, Option Lenght care include lungimea totala in biti a optiunii, si datele asociate optiounii.

Optiuni IPXCP :

|  |  |  |  |
| --- | --- | --- | --- |
| **Nume optiune** | **Tip optiune** | **Lungime optiune** | **Descriere** |
| Numar retea IPX | 1 | 6 | Numarul de retea IPX pentru client. |
| Numar nod IPX | 2 | 6 | Numarul de nod IPX pentru client. |

ATCP

AppleTalk Control Protocol (ATCP), așa cum este folosit de către peer-urile Microsoft PPP este documentat în RFC 1378. ATCP negociază parametrii AppleTalk pentru a configura dinamic un PPP AppleTalk într-o legătură punct-la-punct. Opțiuni comune ATCP includ adresa AppleTalk și informații de server.

ATCP utilizează PPP Protocol ID-ul de  0x80-29. Structura pachet de ATCP este exact la fel cu LCP, cu excepția ca numai tipurile de pachete de la 1 la 7 sunt definite. Pentru pachetele Configure-Request, Configure-Ack, Configure-Nack, and Configure-Reject , porțiunea de date a pachetului ATCP  constă din una sau mai multe opțiuni ATCP . Fiecare opțiune ATCP  constă dintr-un câmp de tip Option Type Field, Option Lenght care include lungimea totala in biti a optiunii, si datele asociate optiounii.

|  |  |  |  |
| --- | --- | --- | --- |
| **Nume optiune** | **Tip optiune** | **Lungime optiune** | **Descriere** |
| Adresa AppleTalk | 1 | 6 | Negociaza numerele de nod si retea AppleTalk. |
| Informatii server | 3 | 16 | Contine informatii despre server. |

NBFCP

NetBIOS Rame Control Protocol (NBFCP), așa cum este folosit de către peer-urile Microsoft PPP este documentat în RFC 2097. NBFCP negociază parametrii NetBEUI pentru a conficura dinamic un PPP NetBEUI intr-o legătură punct-la-punct. Opțiuni comune NBFCP includ opțiuni de filtrare multicast și informații peer.

NBFCP utilizează PPP Protocol ID-ul de  0x80-29. Structura pachet de NBFCP este exact la fel cu LCP, cu excepția ca numai tipurile de pachete de la 1 la 7 sunt definite. Pentru pachetele Configure-Request, Configure-Ack, Configure-Nack, and Configure-Reject , porțiunea de date a pachetului NBFCP constă din una sau mai multe opțiuni NBFCP. Fiecare opțiune NBFCP constă dintr-un câmp de tip Option Type Field, Option Lenght care include lungimea totala in biti a optiunii, si datele asociate optiounii.

|  |  |  |  |
| --- | --- | --- | --- |
| **Nume optiune** | **Tip Optiune** | **Lungime Optiune** | **Descriere** |
| Filtrare Multicast | 3 | 5 | Negociaza transmiterea pachetelor multicast. |
| Informatii Peer | 2 | 17 | Informatii de configurare NetBIOS. |

1. **Bibliografie :**

* [www.wikipedia.org](http://www.wikipedia.org)
* <http://searchnetworking.techtarget.com/definition>
* <http://my.safaribooksonline.com>
* <http://technet.microsoft.com/en-us/library/cc957981.aspx>
* Retele de calculatoare EDITIA A PATRA , Andrew S. Tanenbaum