

Informaciona bezbednost

Bezbedna komunikacija u TCP/IP mrežama

dr Milan Stojkov

Katedra za informatiku

2022.



Fakultet tehničkih nauka
Univerzitet u Novom Sadu

TCP/IP stek i bezbednost

- Sigurna komunikacija može da se obezbedi na različitim nivoima TCP/IP steka
- Potrebne usluge:
 - Poverljivost
 - Neporecivost
 - Integritet
 - Autentifikacija
 - Autorizacija
 - Upravljanje ključevima (generisanje, čuvanje, razmena)

Implementacija bezbednosti na aplikativnom nivou

- Dobre osobine:
 - Implementacija u krajnjim tačkama komunikacije – računarima
 - Aplikacija ne mora da se oslanja na sigurnosne servise operativnog sistema
 - Kompletan pristup podacima koji se štite
 - Jednostavan pristup akreditivima korisnika (npr. tajni ključ)
- Loše osobine:
 - Potrebna je implementacija za svaku aplikaciju posebno
 - Komplikovana izmena postojećih aplikacija
 - Velika verovatnoća pravljenja greške
- Primer – Pretty Good Privacy (PGP)
 - Email klijent se proširuje funkcijama za pronalaženje javnih ključeva, šifrovanje, dešifrovanje, proveru autentičnosti poruka

Implementacija bezbednosti na transportnom nivou

- Dobre osobine:
 - Implementacija u krajnjim tačkama komunikacije – računarima
 - Ne mora se modifikovati svaka aplikacija
 - Kompletan pristup podacima koji se štite
 - Sve aplikacije koriste isti stepen sigurnosti
- Loše osobine:
 - (Neznatna) izmena postojećih aplikacija – zahtevanje sigurnosnih usluga od transportnog sloja
- Primer – Transport Layer Security (TLS)
 - Usluge provere identiteta, integriteta i poverljivosti preko TCP protokola
 - Ne može i za UDP, jer UDP ne održava kontekst tekuće veze

Implementacija bezbednosti na mrežnom nivou

- Dobre osobine:
 - Još manje izmene u aplikacijama
 - Svi transportni protokoli koriste istu infrastrukturu
 - Mogućnost pravljenja virtuelnih privatnih mreža (VPN)
- Loše osobine:
 - Teško je obezbediti uslugu neporecivosti (mnogo lakše na višim slojevima)
 - Teško je obezbediti kontrolu na nivou korisnika na višekorisničkom računaru
- Primer – IPSec (IP Security)
 - Usluge provere identiteta, integriteta i poverljivosti preko TCP protokola

Implementacija bezbednosti na nivou veze

- Ako postoji namenska veza između dva uređaja na mreži (računara, rutera)
- Ako sav saobraćaj mora da se šifruje
- Dobre osobine:
 - Hardverski uređaj za šifrovanje
 - Velika brzina rada
- Loše osobine:
 - Samo za namenske veze – ukoliko su učesnici fizički povezani
- Primer
 - Veza bankomata sa centralom putem namenske veze (iznajmljena linija)

PGP

- PGP = Pretty Good Privacy
- Nastao 1991. kao reakcija svog autora Filipa Cimermana na predlog zakona koji američkoj vladi omogućava pristup otvorenom tekstu svih poruka a koji je primenjiv na proizvođače opreme i komunikacione provajdere
- U to vreme je postojao zahtev da DoD (Department of Defense) mora da odobri patent za novi kriptografski sistem ako sistem koristi ključeve veće od 40 bita
- Tužba od strane RSA Data Security zbog neplaćanja licence za korišćene patente
- Prva legalna verzija PGP-a izvan SAD: 1997.
 - Izvorni kod PGP-a je izvezen u obliku odštampane knjige (to nije zabranjeno - sloboda govora!)
 - Potom OCR-om vraćen u elektronski oblik

PGP

- PGP je protokol koji kombinuje simetrično i asimetrično šifrovanje
- Namena PGP-a je zaštita elektronske pošte
- Koraci:
 - Ključ za šifrovanje od 128 bita se prenosi asimetričnim algoritmom (ElGamal, RSA)
 - Taj ključ postaje ključ za sesiju koji se koristi u simetričnom algoritmu koji šifruje poruku koja se deli na blokove od 64 bita (DES, 3DES, IDEA i AES)
 - Simetrični ključ se šifruje RSA algoritmom pomoću javnog ključa primaoca
 - Primaocu se šalje šifrat otvorenog teksta i šifrat simetričnog ključa

PGP

- Digitalno potpisivanje poruka putem PGP-a
 - Računa se MD5 heš otvorenog teksta
 - Heš se potpisuje privatnim ključem pošiljaoca
 - Pošiljalac šalje otvoreni tekst i potpisani heš primaocu

PGP

- Repozitorijumi javnih ključeva - key servers
 - wwwkeys.pgp.net
 - wwwkeys.eu.pgp.net
 - wwwkeys.us.pgp.net
 - keyserver.ubuntu.com
 - keys.openpgp.org
 - ...
- Poverenje u javne ključeve: "web of trust" umesto stroge CA hijerarhije
 - Vremenom će svaki korisnik prikupiti ključeve drugih ljudi kojima veruje
 - Svoj ključ će publikovati uz ključeve ljudi kojima veruje
 - Primalac će možda prihvatiti da veruje nekom od tih ključeva
 - ... decentralizovana mreža poverenja otporna na otkaze
- Problem slepog prihvatanja ključeva nije rešen za zadovoljavajući način

PGP

- Bezbednost PGP-a zavisi od njegove najslabije komponente:
 - Izbor simetričnog algoritma: DES (slabo), IDEA, Blowfish, AES
 - Izbor asimetričnog algoritma: RSA se smatra sigurnim
 - Izbor heš funkcije: MD5 – loša

PGP

- OpenPGP - rezultat standardizacije u okviru IETF – RFC 2440
 - Definiše standardne formate šifrovanih poruka, potpisa i sertifikata
- Otvorene implementacije:
 - GnuPG
 - Enigmail plugin za Mozilla Thunderbird
 - KMail

SSL

- Komunikacioni protokol razvijen sa ciljem da podrži
 - Kriptografsku bezbednost
 - Interoperabilnost
- Implementacije različitih proizvođača
 - Proširivost
- Različitim kriptografskim algoritmima
 - Relativnu efikasnost
- Optimizuje zauzeće procesora i mrežni protok
 - Keširanjem komunikacionih parametara za uspostavljene veze
- SSL = Secure Sockets Layer
 - Prozvod Netscape-a
 - SSL v2 – prva prihvaćena verzija, imala je bezbednosnih nedostataka
 - SSL v3 – de facto standard od 1996, nikad nije zvanično standardizovan

TLS

- TLS = Transport Layer Security
 - Standardizacija SSL protokola u okviru IETF
 - RFC 2246
 - Podrška u savremenim browserima
 - TLS 1.2 klik za grafički prikaz
 - TLS 1.3 klik za grafički prikaz
- Dva sloja:
 - Record Protocol
 - Handshake Protocol / Alert Protocol / Change Cipher Spec Protocol

Handshake Protocol	Change Cipher Spec Protocol	Alert Protocol	HTTP	Heartbeat Protocol
Record Protocol				
TCP				
IP				

TLS Record Protocol

- TLS Record Protocol:
 - Oslanja se na TCP i daje podršku za protokole višeg nivoa
 - Koristi simetrične algoritme za šifrovanje (poverljivost)
 - Prenos poruka obuhvata i proveru integriteta pomoću hash funkcija

TLS Record Protocol

- Protokol nižeg nivoa koji omogućava prenos poruka drugih protokola:
 - Handshake protocol
 - Alert protocol
 - Change Cipher Spec protocol
 - Heartbeat protocol
 - Protokol aplikativnog nivoa (npr. HTTP)

TLS Record Protocol

- TLS Record protokol prati stanje konekcije
- Stanje se sastoji iz:
 - Izabranog algoritma za kompresiju
 - Izabranog algoritma za šifrovanje
 - Izabranog heša za proveru integriteta poruka
 - Parametara ovih algoritama

TLS Record Protocol

- Slanje poruke:

- 1 Poruka se podeli na fragmente od 2^{14} bajta ili manje
- 2 Fragment se kompresuje (opciono)
- 3 Fragmentu se dodaje hash (MAC – *Message Authentication Code*)
- 4 Fragment se šifrjuje simetričnim algoritmom i *session* ključem
- 5 Na poruku se dodaje zaglavlje koje uključuje verziju i podatke o dužini polja

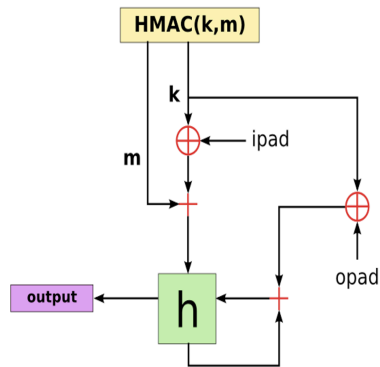
- Struktura:

type	version	length	
data			
MAC			
pad			pad length

type:
 20 - ChangeCipherSpec
 21 - Alert
 22 - Handshake
 23 - Application protocol

TLS Record Protocol

- **MAC = Message Authentication Code**
 - Provera integriteta i provera autentičnosti pošiljaoca
 - Ulazna poruka + ključ → vrednost fiksne dužine
- Konstrukcija pomoću heš funkcije: HMAC
 - h : heš funkcija
 - k : tajni ključ dopunjen nulama do dužine bloka heš funkcije
 - m : poruka
 - $||$: konkatencija
 - \oplus : XOR
 - $opad$: 0x5c5c5c... (u dužini bloka)
 - $ipad$: 0x363636... (u dužini bloka)
- $HMAC_k(m) = h((K \oplus opad) || h((K \oplus ipad) || m))$



TLS Handshake Protocol

- TLS Handshake Protocol:
 - Autentifikacija klijenta i servera i dogovor oko korišćenih algoritama i ključeva
 - Provera identiteta pomoću asimetričnih algoritama
 - Dogovor oko *session* ključa je siguran od prisluškivanja

TLS Handshake Protocol

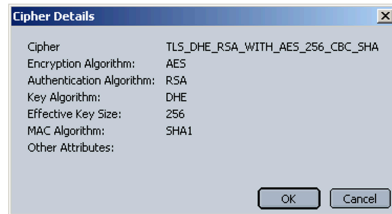
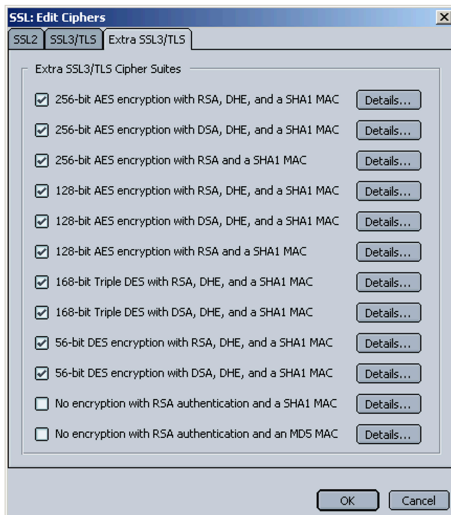
- Namenjen za uspostavljanje komunikacione sesije
- Sesija je skup parametara
- Identifikator sesije
 - Niz bajtova koji jedinstveno identifikuje sesiju (dogovaraju ga klijent i server)
- Potvrda identiteta drugog učesnika u komunikaciji
 - X.509.v3 sertifikat
- Algoritam za kompresiju
 - Kompresija podataka, ako se koristi, vrši se pre šifrovanja
- Cipher spec
 - Simetrični algoritam
 - Heš funkcija (za MAC)
- Master secret
 - 48-bajtni tajni niz koga dele klijent i server – koristi se za generisanje simetričnih ključeva
- Resumable
 - Da li se sesija može koristiti za uspostavljanje novih konekcija
- Jedna sesija može da sadrži više konekcija

TLS Handshake Protocol

- Cipher suite: skup kriptografskih protokola korišćen u komunikaciji

<i>CipherSuite</i>	<i>Key Exchange</i>	<i>Cipher</i>	<i>Hash</i>
TLS_NULL_WITH_NULL_NULL	NULL	NULL	NULL
TLS_RSA_WITH_NULL_MD5	RSA	NULL	MD5
TLS_RSA_WITH_NULL_SHA	RSA	NULL	SHA
TLS_RSA_EXPORT_WITH_RC4_40_MD5	RSA_EXPORT	RC4_40	MD5
TLS_RSA_WITH_RC4_128_MD5	RSA	RC4_128	MD5
TLS_RSA_WITH_RC4_128_SHA	RSA	RC4_128	SHA
TLS_RSA_EXPORT_WITH_RC2_CBC_40_MD5	RSA_EXPORT	RC2_CBC_40	MD5
TLS_RSA_WITH_IDEA_CBC_SHA	RSA	IDEA_CBC	SHA
TLS_RSA_EXPORT_WITH_DES40_CBC_SHA	RSA_EXPORT	DES40_CBC	SHA
TLS_RSA_WITH_DES_CBC_SHA	RSA	DES_CBC	SHA
TLS_RSA_WITH_3DES_EDE_CBC_SHA	RSA	3DES_EDE_CBC	SHA
TLS_DH_DSS_EXPORT_WITH_DES40_CBC_SHA	DH_DSS_EXPORT	DES40_CBC	SHA
TLS_DH_DSS_WITH_DES_CBC_SHA	DH_DSS	DES_CBC	SHA
TLS_DH_DSS_WITH_3DES_EDE_CBC_SHA	DH_DSS	3DES_EDE_CBC	SHA
TLS_DH_RSA_EXPORT_WITH_DES40_CBC_SHA	DH_RSA_EXPORT	DES40_CBC	SHA
TLS_DH_RSA_WITH_DES_CBC_SHA	DH_RSA	DES_CBC	SHA
TLS_DH_RSA_WITH_3DES_EDE_CBC_SHA	DH_RSA	3DES_EDE_CBC	SHA
TLS_DHE_DSS_EXPORT_WITH_DES40_CBC_SHA	DHE_DSS_EXPORT	DES40_CBC	SHA
TLS_DHE_DSS_WITH_DES_CBC_SHA	DHE_DSS	DES_CBC	SHA
TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA	DHE_DSS	3DES_EDE_CBC	SHA
TLS_DHE_RSA_EXPORT_WITH_DES40_CBC_SHA	DHE_RSA_EXPORT	DES40_CBC	SHA
TLS_DHE_RSA_WITH_DES_CBC_SHA	DHE_RSA	DES_CBC	SHA
TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA	DHE_RSA	3DES_EDE_CBC	SHA
TLS_DH_anon_EXPORT_WITH_RC4_40_MD5	DH_anon_EXPORT	RC4_40	MD5
TLS_DH_anon_WITH_RC4_128_MD5	DH_anon	RC4_128	MD5
TLS_DH_anon_EXPORT_WITH_DES40_CBC_SHA	DH_anon	DES40_CBC	SHA
TLS_DH_anon_WITH_DES_CBC_SHA	DH_anon	DES_CBC	SHA
TLS_DH_anon_WITH_3DES_EDE_CBC_SHA	DH_anon	3DES_EDE_CBC	SHA

TLS Handshake Protocol



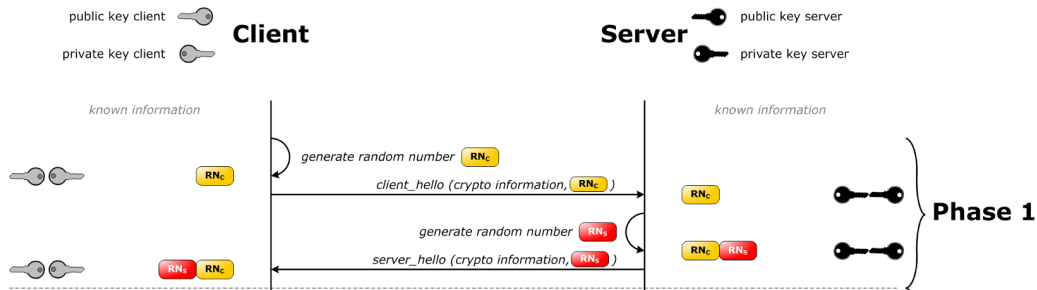
TLS Handshake Protocol

- Inicijalno stanje TLS_NULL_WITH_NULL_NULL
- DH_anon – anonimna razmena ključeva
 - Server se ne autentifikuje
 - Nije otporna na *man-in-the-middle* napad
- RSA: sertifikat sadrži RSA javni ključ; potpis sertifikata koristi RSA
- DSS: sertifikat sadrži DSA javni ključ; potpis sertifikata koristi DSA
- DH_RSA: sertifikat sadrži Diffie-Hellman javni ključ; potpis sertifikata koristi RSA
- DH_DSS: sertifikat sadrži Diffie-Hellman javni ključ; potpis sertifikata koristi DSA
 - Fiksni DH parametri, mora ih potpisati CA
- DHE_*: DH parametri su potpisani sertifikatom, a sertifikat je izdao CA
 - Promenljivi DH parametri, potpisuje ih server, a sertifikat za proveru potpisa je izdao CA
 - DH "ephemeral"

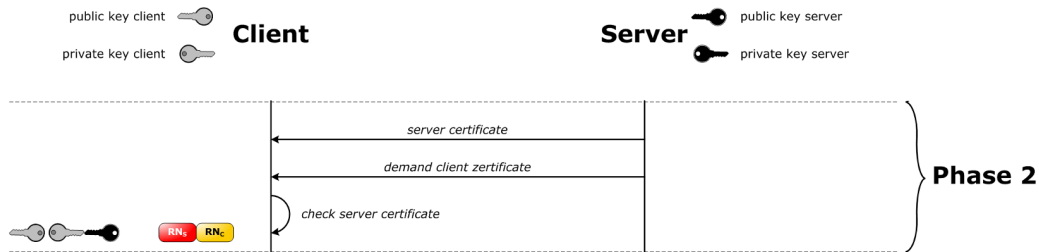
TLS Handshake Protocol

- Tok komunikacije:
 - Klijent otvara vezu i šalje spisak podržanih algoritama i heš funkcija
 - Server od ponuđenih bira najjaču kombinaciju i obaveštava klijenta
 - Server šalje svoju identifikaciju u obliku sertifikata
 - Klijent može kontaktirati CA radi provere sertifikata - to nije obuhvaćeno TLS protokolom!
 - Klijent šifrjuje slučajan broj javnim ključem servera i šalje mu ga
 - Na osnovu slučajnog broja klijent i server generišu session ključ

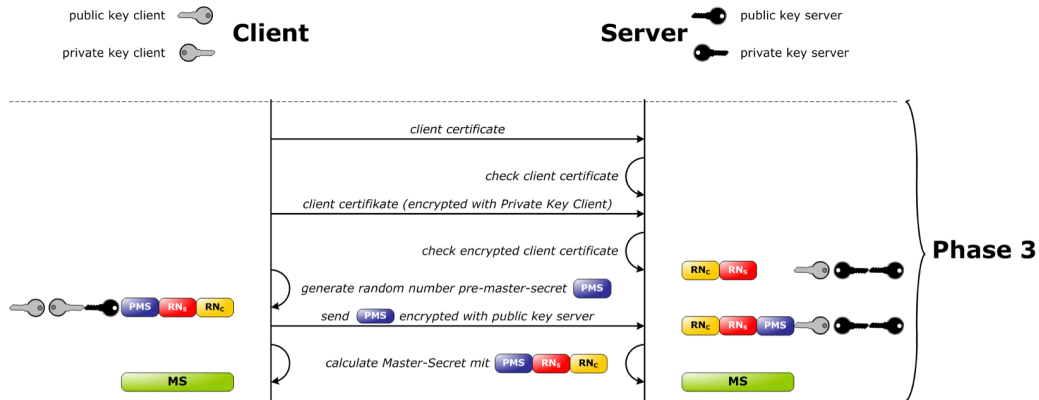
TLS Handshake Protocol





TLS Handshake Protocol



TLS Handshake Protocol





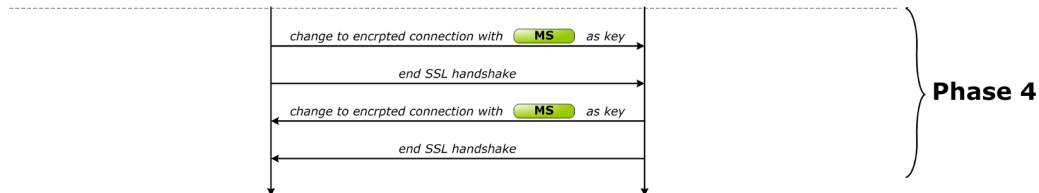
TLS Handshake Protocol

public key client 
private key client 

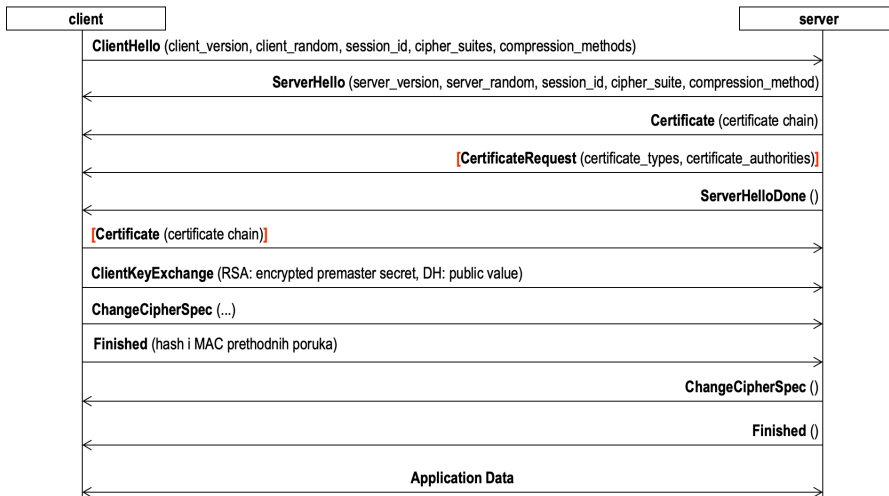
Client

Server

 public key server
 private key server

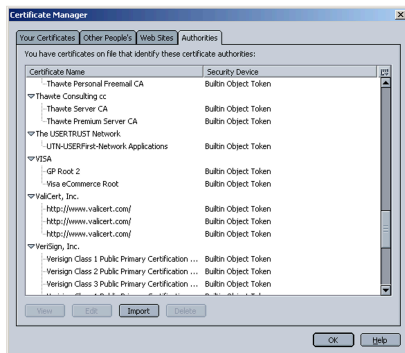


TLS Handshake Protocol - tok komunikacije



Sertifikati

- “Root CA” je self-signed
- Browseri sadrže “root CA” sertifikate
- Svaki sertifikat sadrži “CA flag”
 - Da li vlasnik ima pravo da izdaje nove sertifikate, tj. da li je vlasnik takođe CA



Sertifikati

- Internet Explorer 5.0-6.0 bag
 - Ne proverava da li posrednički sertifikati imaju pravo da izdaju sertifikate
- Primer:
 - Kupimo sertifikat za nastyattacker.com
 - Iskoristimo ga za potpisivanje sertifikata za amazon.com
 - Presrećemo saobraćaj sa amazon.com i podmećemo svoj lažni sertifikat
- Primer kako mali bag može da sruši sistem čija izgradnja košta puno \$M

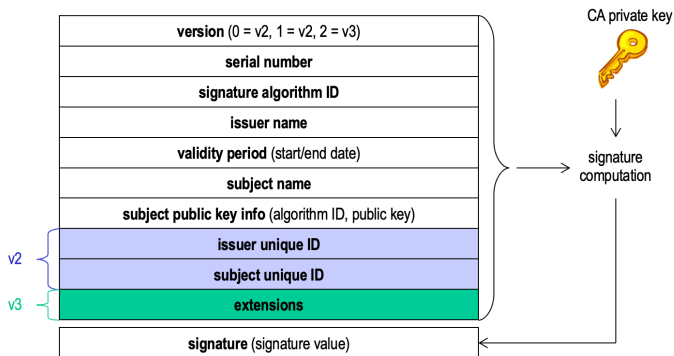
Sertifikati

- Kako postati “root CA”
 - Troškovi oko \$0.5M
 - Finansijski, regulatorni i politički uslovi
- Kupovina sertifikata kod CA
 - Slično reketiranju

“...If you fail to renew your Server ID prior to the expiration date, operating your Web site will become far riskier than normal [...] your Web site visitors will encounter multiple, intimidating warning messages when trying to conduct secure transactions with your site. This will likely impact customer trust and could result in lost business for your site....”

Sertifikati

- X.509 standard



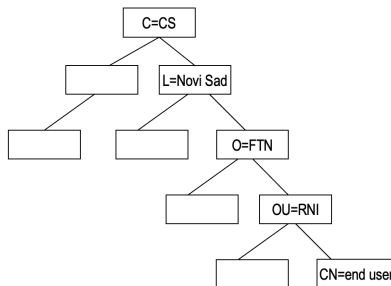
Sertifikati

- X.509 standard
 - Primer sertifikata:

Version: 3 (0x2)
Serial Number: 1 (0x1)
Signature Algorithm: md5WithRSAEncryption
Issuer: C=CS, L=Novi Sad, O=FTN, OU=Odeljenje za sertifikate, CN=FTN CA, Email=ca@ftn.uns.ac.rs
Validity:
 Not Before: Jun 8 10:00:00 2004 GMT
 Not After: Jun 7 10:00:00 2005 GMT
Subject: C=CS, L=Novi Sad, O=FTN, OU=Katedra za informatiku, CNGoran Sladić, Email=sladicg@uns.ac.rs
Subject Public Key Info:
 Public Key Algorithm: rsaEncryption
 RSA Public Key: (1024 bit)
 Modulus (1024 bit): 00:b3:4e:75:76:fc:4c:c3:bd:61:6c:14:41:8f:47:...
 Exponent: 65537 (0x10001)
X.509v3 Extensions:
 X.509v3 Basic Constraints
 CA: false
 Netscape Comment:
 OpenSSL Generated Certificate
 X.509v3 Subject Key Identifier:
 a6:db:b8:78:19:7a:c4:67:23:de:03:a3:ee:d4:26:5e:78:14:71:61
Signature:
 9f:15:a8:cb:6c:a9:0d:d4:61:24:b9:7a:bc:29:e4:29:8b:4c:...

Sertifikati

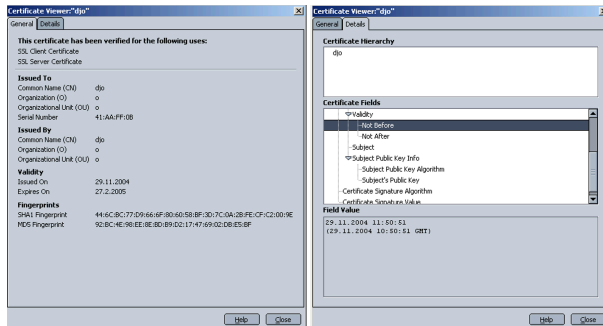
- X.509 standard
 - Hijerarhijska organizacija imena (C=CS, L=Novi Sad, O=FTN, ...) potiče od X.500 standarda, sveobuhvatnog direktorijumskog servisa



- Svaki čvor stabla ima svoj CA

Sertifikati

- X.509 standard
 - Problem distribucije ključeva pretvoren je u problem distribucije imena
- Ljudi sa istim imenom i prezimenom u istoj organizaciji
- Kreiranje jedinstvenih naziva – pretraživanje po imenu više nema smisla
 - John Smith 1 vs John Smith 2 vs John Smith 3



Sertifikati

- X.500 nije zaživeo
 - Organizacija možda ne želi da otkrije svoju internu strukturu
- X.509 sertifikati mogu biti vezani za:
 - X.500 distinguished name (prethodni primer)
 - Alternative name: email adresa, DNS ime
- X.509 obuhvata i standard za CRL
 - Online Certificate Status Protocol (OCSP)

Alert Protocol

- Koristi se za prenošenje upozorenja vezanih za TLS entitetu sa kojim se komunicira
- Poruke upozorenja su komprimovane i šifrovane
- Dva bajta u poruci
- Prvi bajt - *level*:
 - 1: warning - veza ili bezbednost mogu biti nestabilni
 - 2: fatal - veza ili bezbednost mogu biti ugroženi, ili je nastupila neotklonjiva greška i prekida se konekcija
- Drugi bajt - *description*:
 - 0: close notify
 - ...
 - 10: unexpected message
 - ...
 - 20: bad record MAC
 - ...
 - 44: certificate revoked
 - 45: certificate expired

Change Cipher Spec Protocol

- Najjednostavniji od četiri TLS-specifična protokola koji koriste TLS Record Protocol
- Sastoji se od jedne poruke koja se sastoji od jednog bajta sa vrednošću 1
- Jedina svrha ove poruke je da se stanje na čekanju kopira u trenutno stanje, čime se ažurira *cipher suite* koji će se koristiti u trenutnoj konekciji

Heartbeat Protocol

- RFC 6250
- U kontekstu računarskih mreža, *heartbeat* je periodični signal koji generiše hardver ili softver da bi ukazao na normalan rad ili da bi sinhronizovao druge delove sistema
- Heartbeat protokol se obično koristi za praćenje dostupnosti entiteta protokola
- Radi nad TLS Record Protocolom i sastoji se iz dve poruke
 - `heartbeat_request`
 - `heartbeat_response`
- Uspostavlja se tokom faze 1 Handshake protokola
- Svaki entitet pokazuje da li podržava "ping"
- Ako podržava, entitet pokazuje da li je voljan da prima poruke `heartbeat_request` i odgovara porukama `heartbeat_response` ili je voljan da samo šalje poruke `heartbeat_request`

Heartbeat Protocol

- Sadržaj `heartbeat_request` poruke:
 - Dužina poruke
 - Poruka (između 16 bajta i 64 kilobajta)
 - *Padding* (nasumični sadržaj)
- Poruka `heartbeat_response` mora da sadrži tačnu kopiju `heartbeat_request` poruke
- Protokol ima dve uloge:
 - Uverava pošiljaoca poruke da je primalac živ
 - Generiše aktivnost u komunikaciji dok je ona besposlena (*idle*) kako bi odvratila *firewall* da prekine konekciju

Napadi na SSL/TLS

- Napadi na Handshake protokol
 - 1998. godine predstavljen je pristup kompromitovanju protokola zasnovan na iskorišćavanju formatiranja i implementacije RSA (Bleichenbacher, D. Chosen Ciphertext Attacks against Protocols Based on the RSA Encryption Standard PKCS #1. CRYPTO '98, 1998)
 - Napad je poboljšan i prilagođen ne samo da spreči kontramere, već i da ubrza napad (Bardou R. et al. Efficient Padding Oracle Attacks on Cryptographic Hardware. INRIA, Rapport de recherche RR-7944, April 2012)

Napadi na SSL/TLS

- Napadi na podatke i application data protokole
 - Browser Exploit Against SSL/TLS (BEAST) – Duong T, Rizzo J. Here come the ☯ Ninjas. May 2011
 - Padding Oracle On Downgraded Legacy Encryption (POODLE) – Möller B. et al. This POODLE Bites: Exploiting The SSL 3.0 Fallback. Google. September 2014

Napadi na SSL/TLS

- Napadi na PKI
 - Provera validnosti X.509 sertifikata je podložna raznim napadima
 - Često korišćene biblioteke za SSL/TLS pate od ranjivih implementacija validacije sertifikata
 - Postoje slabosti u izvornom kodu OpenSSL, GnuTLS, JSSE, ApacheHttpClient, Weberknecht, cURL, PHP, Python bibliotekama (Georgiev M. et al. The Most Dangerous Code in the World: Validating SSL Certificates in Non-Browser Software. ACM Conference on Computer and Communications Security, 2012)

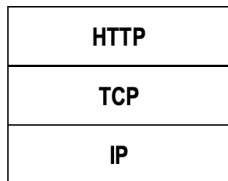
Heartbleed

- Greška otkrivena 2014. u OpenSSL implementaciji Heartbeat protokola
- Ranjivost nije mana dizajna u TLS specifikaciji već je to programska greška u biblioteci OpenSSL
- Pre nego što je greška popravljena, OpenSSL verzija Heartbeat protokola je radila na sledeći način:
 - Softver čita zahtev i dodeljuje bafer dovoljno veliki da sadrži zaglavlje poruke, podatke i *padding*
 - Prepisuje trenutni sadržaj bafera sa dolaznom porukom iz zahteva, menja prvi bajt da bi ukazao na tip odgovora, a zatim prenosi odgovor, koji uključuje polje dužine poruke i poruku
 - Softver ne proverava da li se dužina poruke i poruka slažu!
 - Napadač može da pošalje poruku koja ima informaciju o maksimalnoj dužini poruke (64 KB), ali šalje poruku koja ima najmanju moguću dužinu (16 bajtova)
 - To znači da skoro 64 KB podataka u baferu nije prepisano novim informacijama i sve što se desilo u memoriji u tom trenutku biće poslato kao odgovor

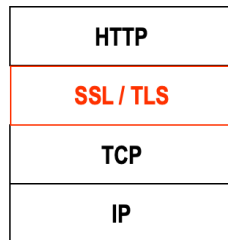
HTTPS

- Bezbedna komunikacija putem HTTP protokola
- Sam HTTP protokol se ne menja, već se on oslanja na drugi protokol koji omogućava bezbednu komunikaciju (HTTP over SSL)

http://



https://



HTTPS

- Razlika koju vidi krajnji korisnik u web pretraživaču je u URL adresi koja počinje sa https:// umesto sa http://
- Obična HTTP konekcija koristi port 80, HTTPS koristi port 443
- Kada se koristi HTTPS sledeći elementi komunikacije su šifrovani:
 - URL traženog dokumenta
 - Sadržaj dokumenta
 - Sadržaj forme koju popunjava korisnik
 - Kolačići (cookies) koji se šalju od pretraživača ka serveru i obrnuto
 - Sadržaj HTTP zaglavlja

HTTPS

- Započinjanje konekcije:
 - Agent koji je u ulozi HTTP klijenta je takođe i u ulozi TLS klijenta
 - Klijent inicira konekciju ka serveru na odgovarajućem portu i šalje TLS ClientHello da započne *TLS handshake*
 - Kada se *TLS handshake* završi klijent započinje svoj prvi HTTP zahtev

HTTPS

- Zatvaranje konekcije:
 - HTTP klijent ili server mogu da označe zatvaranje konekcije sa `Connection: close`
 - Zatvaranje HTTPS konekcije zahteva da TLS zatvori konekciju sa entitetom sa druge strane koji će započeti i zatvaranje TCP konekcije
 - Na TLS nivou svaka strana koristi TLS alert prtokol da razmeni `close_notify` poruku
 - TLS može da zatvori konekciju kada pošalje svoju `close_notify` poruku bez čekanja da druga strana uradi isto
 - Ovo može da se uradi samo ako je strana koja zatvara konekciju sigurna da je primila sve potrebne poruke
 - HTTP klijenti treba da budu spremni i na scenarija kada je TCP konekcija zatvorena bez prethodnog `close_notify` alerta ili `Connection: close` indikatora

IPSec

- Skup proširenja IPv4 koji obezbeđuje privatnost, integritet, proveru identiteta i neporecivost
- Integralni deo IPv6
- Na mrežnom sloju TCP/IP steka

IPSec

- Koristi sledeće komponente
 - Diffie-Hellman za razmenu ključeva
 - Algoritme za digitalno potpisivanje komunikacije pri DH razmeni ključeva
- Radi potvrde identiteta učesnika u komunikaciji – sprečava se *man-in-the-middle* napad
 - DES, 3DES, AES za šifrovanje
 - MD5 i SHA kao osnova za HMAC funkcije
 - Sertifikate koje potpisuje CA

IPSec

- Dva nezavisna protokola:
 - AH (*Authentication Header*)
 - Usluge integriteta, provere identiteta i neporecivosti
 - ESP (*Encapsulated Security Payload*)
 - Integritet, identitet, neporecivost i poverljivost podataka

IPSec / Authentication Header

- RFC 2402
- AH zaglavlje se smešta između IP zaglavlja i podataka koji slede
- Ne enkapsulira podatke iz protokola kojima pruža uslugu!

IPSec / Authentication Header

- Polja u AH zaglavlju:

- *next header* – tip podataka koji sledi posle AH zaglavlja (npr. 6 - TCP, 17 - UDP, 50 - ESP)
- *payload length* – dužina podataka u 32-bitnim rečima umanjena za 2
- *reserved* – 16 bita rezervisano za buduće potrebe, vrednost 0
- *security parameters index* – skup parametara veze koji se definišu prilikom uspostave veze
- *sequence number* – povećava se prilikom svakog slanja paketa sa istim parametrima veze
 - Zaštita od napada ponavljanjem paketa
- *authentication data* – vrednost na osnovu koje se proverava integritet i autentičnost
 - MAC vrednost IP zaglavlja, AH zaglavlja postavljenog na 0, i svih podataka protokola višeg sloja

IPSec / Authentication Header

next header	payload length	reserved
security parameters index		
sequence number		
authentication data		

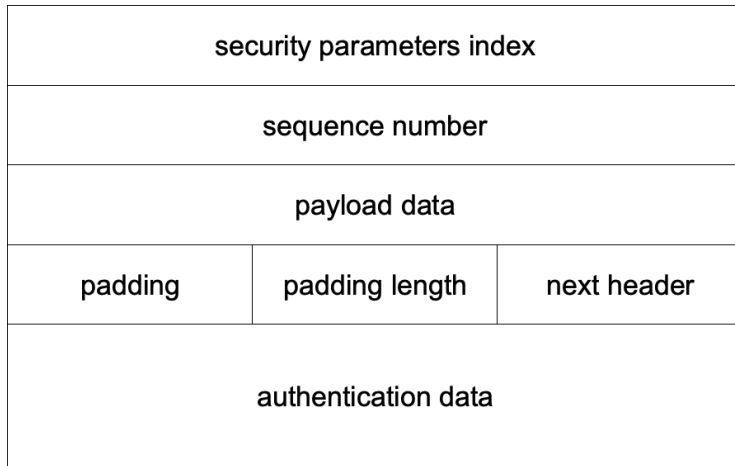
IPSec / Encapsulated Security Payload

- RFC 2406
- Smešta se posle IP zaglavlja
- Enkapsulira sve podatke iz protokola višeg sloja
- Dodaje završni slog u koji se mogu smestiti podaci za proveru identiteta

IPSec / Encapsulated Security Payload

- Polja u EPS zaglavlju:
 - *security parameters index* – skup parametara veze, isto kao kod AH
 - *sequence number* – brojač paketa sa istim parametrima veze, isto kao kod AH
 - *payload data* – podaci iz protokola višeg sloja i
 - *padding* – dopuna paketa (zbog šifrovanja blokova fiksne dužine ili zbog razloga implementacije)
 - *padding length* – dužina dopune
 - *next header* – tip podataka koji sledi posle ESP zaglavlja, isto kao kod AH
 - *authentication data* – samo kada se koristi provera identiteta; MAC se računa na osnovu celog ESP paketa osim ovog polja

IPSec / Encapsulated Security Payload

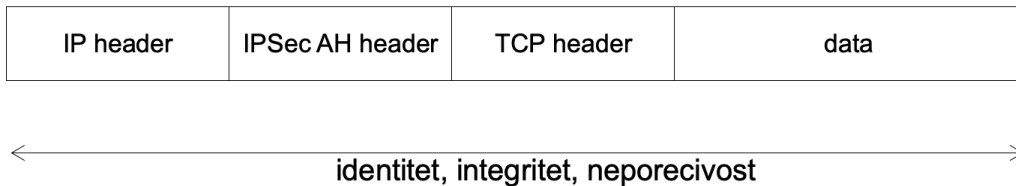


IPSec

- Dva režima rada:
 - Transportni režim
 - Šifruju se samo podaci u IP paketu, dok se IP zaglavlje ne menja
 - Svakom paketu se dodaje samo nekoliko okteta
 - Ruteri vide *source* i *destination IP*
 - Tunelovanje
 - Poseban oblik IP paketa
 - Tunel čine klijent i server koji su konfigurisani da koriste IPSec tunelovanje
 - Unapred dogovoreni mehanizmi za enkapsulaciju i šifrovanje kompletnih IP paketa
 - Siguran prenos preko javnih ili privatnih mreža

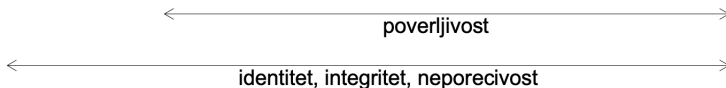
IPSec / transportni režim

- Ako se koristi AH



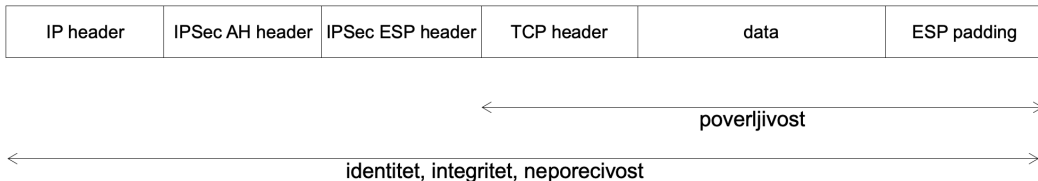
IPSec / transportni režim

- Ako se koristi ESP
 - Svi podaci iz višeg sloja su šifrovani
 - ESP proverava integritet svog zaglavlja i podataka, ali ne i IP zaglavlja
 - Moguće izmene IP zaglavlja



IPSec / transportni režim

- Ako se koristi ESP + AH
 - AH za integritet, identitet i neporecivost celog IP paketa
 - ESP za poverljivost
 - Prvo se formira ESP deo, potom AH
 - ESP ne sadrži polje auth data, već to radi AH



IPSec / tunelovanje

- Sigurna komunikacija *gateway-to-gateway* između dve mreže
- VPN (*Virtual Private Network*)
- U *gateway-to-gateway* varijanti krajnji čvorovi u komunikaciji ne moraju podržavati IPSec
 - Moguća je i komunikacija *računar-gateway* ili *računar-računar*, tada moraju podržavati IPSec
- Formira se novi IP paket koji enkapsulira originalan IP paket

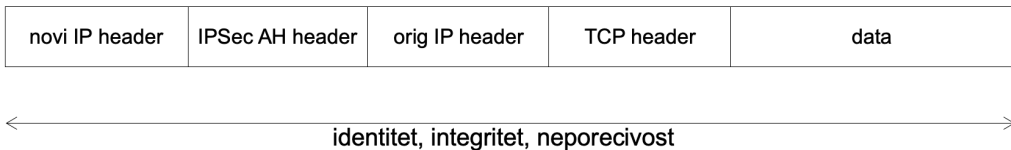


IPSec / tunelovanje

- Tok komunikacije:
 - Pošiljalac formira IP paket i šalje ga svom *gateway*-u
 - *Gateway* enkapsulira primljeni paket u novi paket (po RFC 2003) i formira AH i ESP zaglavlja
 - Tako formirani novi paket se šalje drugom *gateway*-u
 - Tamo se uklone dodatna zaglavlja, (ako treba) dešifruje paket i proveriti njegov integritet
 - Originalni IP paket se isporučuje odredištu

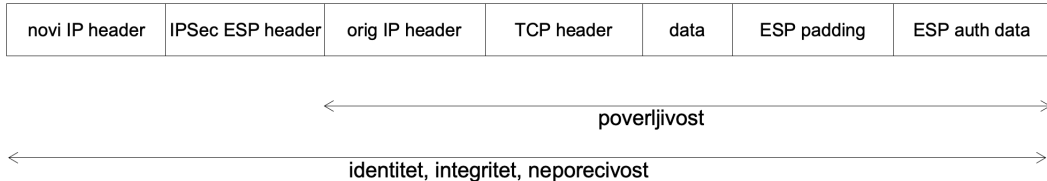
IPSec / tunelovanje

- Ako se koristi AH
 - Integritet, identitet, neporecivost
 - Originalni IP paket se enkapsulira u novi kome se dodaje AH zaglavlje



IPSec / tunelovanje

- Ako se koristi ESP
 - Integritet, identitet, neporecivost i poverljivost
 - Šifruje se ceo enkapsulirani IP paket



IPSec / tunelovanje

- Ako se koristi AH + ESP
 - Nije predviđeno po RFC 2401

IPSec / uspostava veze

- IPSec ne definiše mehanizam za uspostavljanje parametara veze
- Protokoli zasnovani na Diffie-Hellman algoritmu:
 - Photuris – RFC 2522
 - SKIP (Simple Key Management for Internet Protocols) – Draft
- Rašireniji postupci
 - ISAKMP (Internet Security Association and Key Management Protocol) – RFC 2408
 - IKE (Internet Key Exchange)

IKE

- RFC 2409
- Kombinuje:
 - ISAKMP: infrastruktura za proveru identiteta i razmenu ključeva
 - Oakley RFC 2412: način razmene ključeva
 - SKEME: način razmene ključeva i obezbeđuje anonimnost
- Uspostava veze po IKE ima dve faze:
 - Uspostavljanje IKE SA (Security Association) parametara
 - Uspostavljanje IPSec SA parametara

IKE / uspostavljanje IKE SA

- Parametri IKE veze (SA)
 - Algoritam za šifrovanje
 - Heš funkcija
 - Metoda provere identiteta
 - RSA/DSA digitalni potpisi
 - Tajni ključ (preshared key)
 - Puna PKI infrastruktura (eliminiše *man-in-the-middle* napad)
 - Oakley grupa koja definiše DH razmenu ključeva (RSA ili eliptične krive)

IKE / uspostavljanje IKE SA

- Dva režima rada:
 - *Main mode*
 - Zaštita identiteta učesnika u komunikaciji
 - Razmenjuje se šest poruka tokom uspostave IKE SA
 - *Aggressive mode*
 - Nema zaštite identiteta učesnika
 - Razmenjuju se tri poruke – brža uspostava veze

IKE / uspostavljanje IKE SA

- Ključevi u IKE:
 - Glavni ključ koji se koristi za generisanje ostalih ključeva
 - Ključ koji IKE SA koristi za šifrovanje poruka
 - Ključ koji IKE SA koristi za proveru identiteta i integriteta
 - Ključ koji služi za generisanje IPSec SA
- *Cookies*: heš vrednost na osnovu
 - IP adresa, port, protokol, *timestamp*, *secret value*

IKE / uspostavljanje IPSec SA

- Izvodi se u *quick mode*
 - Koristi se prethodno uspostavljen IKE SA skup parametara
 - IPSec SA se određuje na osnovu IKE SA

IPSec i potrošnja resursa

- Dodatno procesorsko vreme za kriptografske operacije
- Povećan mrežni saobraćaj
 - Dodatna zaglavlja
 - *Padding*
 - Inicijalizacioni vektor za šifrovanje u CBC režimu