SSL protokol

Za zaštitu komunikacije izme?u klijenta i servera npr. *web* klijenta (*web* pretraživa?a) i *web* servera, na?eš?e se koristi SSL (*Secure Sockets Layer*) protokol, odnosno njegova standardizovana verzija TLS (*Transport Layer Security*). SSL protokol nastao je u Netscapu, ali je standardizovan u IETF-u pod nazivom Transport Layer Security (TLS). Naziv SSL se zadržao, pa ?emo ga i mi ?eš?e koristiti. SSL može biti ugra?en u softverski paket (npr. Microsoft Explorer pretraživa? dolazi sa ugra?enim SSL protokolom i ve?ina *web* servera ima implementiran protokol). Alternativno SSL/TLS može biti instaliran kao deo TCP/IP protokol steka i tako transparentan za aplikativni nivo.

SSL (*Secure Sockets Layer*) protokol funkcioniše po klijent-server modelu. Klijent je strana koja inicira sigurnu komunikaciju, dok server odgovara na zahtev klijenta. Naj?eš?i primer koriš?enja SSL protokola je *https* koji predstavlja sigurnu verziju *http*-a (*secure http*) i koristi se za uspostavu zašti?ene veze sa nekim *web* serverom, naj?eš?e za potrebe elektronskog pla?anja. U tom slu?ajum, *web* pretraživa? predstavlja SSL klijenta, a *web* server, odnosno sajt kome se pristupa, je SSL server.

Sa stanovišta SSL protokola, ono što pravi razliku izme?u klijenta i servera su akcije koje oni preduzimaju prilikom pregovora oko sigurnosnih parametara. Klijent inicira komunikaciju, porukom u kojoj se nalazi i njegov predloži skup SSL opcija koje ?e se koristiti za uspostavu zašti?enog kanala. Server, na osnovu onoga što je klijent ponudio, bira skup opcija koje ?e se koristiti. Iako je kona?na odluka na serveru, server može da bira samo iz skupa parametara koje je klijent inicijalno predložio.

B.1 SSL poruke

Najkompleksniji deo SSL protokola je deo oko zapo?injanja i dogovora o parametrima komunikacije izme?u klijenta i servera, *SSL Handshake*. On omogu?ava serveru i klijentu da me?usobno autentifikuju jedan drugog, da izaberu algoritam za šifrovanje, MAC algoritam i da razmene kriptografske klju?eve koji ?e se koristiti za zaštitu podataka u SSL zapisu.

SSL Handshake se sastoji iz tri pod-protokola: *SSL Handshake*, *SSL Change Cipher Spec* i *SSL Alert*.

Poruke koje se razmenjuju prilikom *SSL Handshake* su slede?e:

Alert – Obaveštava drugu stranu o mogu?oj sigurnosnoj pretnji ili o prekidu komunikacije.

ApplicationData – Sami korisni podaci koje dve strane razmenjuju, šifrovani, autentifikovani i/ili verifikovani od strane SSL-a.

Certificate – Poruka koja nosi digitalni sertifikat pošiljaoca (koji sadrži i njegov javni klju?).

CertificateRequest – Zahtev koji server šalje klijentu traže?i mu da pošalje svoj sertifikat.

CertificateVerify – Poruka koju šalje klijent kako bi dokazao da poseduje privatni klju? koji odgovara javnom klju?u koji se nalazi u njegovom sertifikatu.

ChangeCipherSpec – Poruka kojom se ozna?ava po?etak koriš?enja sigurne komunikacije sa prethodno dogovorenim parametrima.

ClientHello – Poruka koju šalje klijent, u kojoj navodi listu sigurnosnih parametra koje podržava i želi da koristi za uspostavu sigurne komunikacije.

ClientKeyExchange – Poruka od klijenta koja sadrži kriptografske klju?eve za uspostavu sigurne komunikacije.

Finished – Potvrda da je inicijalno pregovaranje završeno i da je uspostavljena sigurna komunikacija.

HelloRequest – Zahtev od servera da klijent po?ne (ili restartuje) SSL proces za dogovor oko parametara koji ?e se koristiti.

ServerHello – Poruka od servera koja specificira sigurnosne servise koji ?e se koristiti u komunikaciji.

ServerHelloDone – Potvrda koju šalje server kako bi potvrdio klijentu da je završio sa slanjem svih zahteva klijentu za uspostavu sigurne komunikacije.

ServerKeyExchange - Poruka od servera koja sadrži kriptografske klju?eve za uspostavu sigurne komunikacije.

B.2 Uspostavljanje zašti?ene komunikacije

Osnovna funkcija koju SSL klijent i server mogu da urade je uspostavljanje kanala po kome se odvija sigurna komunikacija.

Na slici B.2.1 je prikazana razmena poruka izme?u SSL klijenta i servera koja prethodi uspostavljanju zašti?ene komunikacije, a u tabeli B.2.1 je dato objašnjenje odgovaraju?ih koraka u razmeni.

Slika B.2.1 SSL koristi 9 poruka prilikom uspostavljanja zašti?ene veze

Razmena poruka prilikom uspostavljanja zašti?ene veze

Poruka	Akcija
1	Klijent šalje *ClientHello* poruku sa predlozima za izbor SSL parametara
2	Server odgovara *ServerHello* porukom sa SSL parametrima koje je izabrao
3	Server šalje informaciju o svom javnom klju?u u *ServerKeyExchange* poruci
4	Server zaklju?uje svoj deo pregovora *ServerHelloDone* porukom
5	Klijent šalje klju? koji ?e se koristiti tokom sesije (kriptovan javnim klju?em servera) u *ClientKeyExchange* poruci
6	Klijent šalje *ChangeCipherSpec* poruku kojom aktivira koriš?enje zašti?ene komunikacije sa dogovorenim parametrima za sve budu?e poruke
7	Klijent šalje *Finished* poruku kako bi server proverio novoaktivirane sigurnosne opcije
8	Server šalje *ChangeCipherSpec* poruku kojom aktivira koriš?enje zašti?ene komunikacije sa dogovorenim parametrima za sve budu?e poruke
9	Server šalje *Finished* poruku kako bi klijent proverio novoaktivirane sigurnosne opcije

Tabela B.2.1 Razmena SSL poruka tokom uspostaljanja sigurnog kanala komunikacije

B.3 Autentifikacija servera

SSL sadrži mehanizme koji omogu?avaju svakoj strani da autentifikuje drugu stranu u komunikaciji.

Autentifikaciju servera, klijent može zahtevati tokom uspostavljanja veze. Na ovaj na?in, korisnik proverava identitet servera sa kojim komunicira, da nije došlo do kra?e identiteta, odnosno, lažnog predstavljanja od strane nekog napada?a.

Kada klijent zahteva autentifikaciju, server mu odgovara *Certificate* porukom umesto porukom *ServerKeyExchange*. *Certificate* poruka sadrži lanac sertifikata koji po?inje sertifikatom samog server, a završava se sertifikatom korenog sertifikacionog tela (root CA). Klijent je dužan da proveri da li može da veruje sertifikatu koji dobije od servera. To uklju?uje proveru lanca poverenja i provoru validnosti sertifikata.

Nakon što je utvrdio identitet servera, klijent nastavlja proceduru opisanu u predhodnom poglavlju. Naravno klju? K, koji klijent šalje serveru je šifrovan javnim klju?em servera koji se nalazi u sertifikatu dobijenom od servera. Klijent je siguran da samo server koji poseduje odgovaraju?i privatni klju?, može da dešifruje poruku klijenta i uspešno nastavi komunikaciju.

Da bi opisani scenario funkcionisao, server mora da ima svoj digitalni sertifikat (koji nazivamo serverski SSL sertifikat) i on mora biti instaliran na serveru.

Na slici B.3.1 je prikazana razmena poruke tokom autentifikacije servera na zahtev klijenta, dok je u tabeli B.3.1 je dato objašnjenje za odgovaraju?e korake u razmeni.

Slika B.3.1 Dve dodatne SSL poruke autentifikuju server

Razmena poruka pri autentifikaciji servera	
Poruka	**Akcija**
1	Klijent šalje *ClientHello* poruku sa predlozima za izbor SSL parametara
2	Server odgovara *ServerHello* porukom sa SSL parametrima koje je izabrao
3	Server šalje *Certificate* poruku koja sadrži sertifikat servera
4	Server zaklju?uje svoj deo pregovora *ServerHelloDone* porukom
5	Klijent šalje klju? koji ?e se koristiti tokom sesije (kriptovan javnim klju?em servera koji se nalazi u sertifikatu dobijenom od servera) u *ClientKeyExchange* poruci
6	Klijent šalje *ChangeCipherSpec* poruku kojom aktivira koriš?enje zašti?ene komunikacije sa dogovorenim parametrima za sve budu?e poruke
7	Klijent šalje *Finished* poruku kako bi server proverio novoaktivirane sigurnosne opcije
8	Server šalje *ChangeCipherSpec* poruku kojom aktivira koriš?enje zašti?ene komunikacije sa dogovorenim parametrima za sve budu?e poruke
9	Server šalje *Finished* poruku kako bi klijent proverio novoaktivirane sigurnosne opcije

Tabela B.3.1 Razmena SSL poruka pri autentifikaciji servera

B.4 Razdvajanje enkripcije od autentifikacije

Nedostatak postupka autentifikacije iz predhodnom poglavlja, je u tome što se isti javni klju? servera koristi za potvrdu identiteta servera i za enkripciju klju?a K koji ?e se koristiti za šifrovanje sadržaja komunikacije tokom trajanja sesije. U nekim slu?ajevima ni ne postoji odgovaraju?a podrška za sprovo?enje opisanog postupka, budu?i da se neki sigurnosni algoritmi (npr. DSA – *Digital Signature Algorithm*) mogu koristiti samo za digitalno potpisivanje poruke, ali ne i za kriptovanje. U takvoj situaciji, nije izvodljivo da poruka sa klju?em K bude šifrovana javnim klju?em server koji se nalazi u sertifikatu kojim je dokazan njegov identitet.

Dakle, potrebno je razdvojiti enkripciju od autentifikacije, pa je server dužan da odgovori i *Certificate* porukom i *ServerKeyExchange* porukom.

Certificate poruka sadrži sertifikat servera. Javni klju? servera koji se nalazi u njegovom sertifikatu koristi se samo kako bi se potvrdio identitet servera, odnosno, izvršila njegova autentifikacija.

ServerKeyExchange poruku sadrži drugi javni klju? servera, koji klijent treba da koristi za enkripciju informacije o klju?evima K koji ?e se koristiti tokom sesije. Razlika je u tome što je informacija o drugom javnom klju?u servera sada može biti potpisana privatnim klju?em servera, ?iji je odgovaraju?i javni klju? predhodno poslat sa njegovim sertifikatom. Tako klijent može da potvrdi da server zaista poseduje privatni klju? koji odgovara javnom klju?u sadržanom u sertifikatu servera.

Na slici B.4.1 je prikazana razmena poruka koja odgovara objašnjenoj situaciji.



Slika B.4.1 Tri dodatne SSL poruke odvajaju autentifikaciju od enkripcije

U tabeli B.4.1 su objašnjeni svi koraci u razmeni poruka.

Razmena poruka pri razdvajanju autentifikacije od enkripcije	
Poruka	**Akcija**
1	Klijent šalje *ClientHello* poruku sa predlozima za izbor SSL parametara
2	Server odgovara *ServerHello* porukom sa SSL parametrima koje je izabrao
3	Server šalje *Certificate* poruku koja sadrži sertifikat servera
4	Server šalje javi klju? koji ?e klijent da koristi za enkripciju u *ServerKeyExchange* poruci; ova poruka je potpisana privatnim klju?em servera
5	Server zaklju?uje svoj deo pregovora *ServerHelloDone* porukom
6	Klijent šalje klju? koji ?e se koristiti tokom sesijd (kriptovan javnim klju?em servera dobijenim u *ServerKeyExchange* poruci) u *ClientKeyExchange* poruci
7	Klijent šalje *ChangeCipherSpec* poruku kojom aktivira koriš?enje zašti?ene komunikacije sa dogovorenim parametrima za sve budu?e poruke
8	Klijent šalje *Finished* poruku kako bi server proverio novoaktivirane sigurnosne opcije
9	Server šalje *ChangeCipherSpec* poruku kojom aktivira koriš?enje zašti?ene komunikacije sa dogovorenim parametrima za sve budu?e poruke
10	Server šalje *Finished* poruku kako bi klijent proverio novoaktivirane sigurnosne opcije

Tabela B.4.1 Razmena SSL poruka pri razdvajanju autentifikacije od enkripcije

B.5 Autentifikacija klijenta

Server tako?e može da zahteva autentifikaciju klijenta, tokom razmene inicijalnih *Hello* poruka izme?u klijenta i servera. SSL specificira da server ne može da traži autentifikaciju klijenta ukoliko se on prethodno nije autentifikovao klijentu.

Napomena: SSL koristi javni klju? klijenta samo za digitalno potpisivanje, odnosno autentifikaciju klijenta. Za razliku od slu?aja sa serverom, ne postoji potreba za enkripcijom koriš?enjem javnog klju?a klijenta.

Na slici B.5.1 je prikazana odgovaraju?a razmena poruka, a u tabeli B.5.1 su objašnjeni odgovaraju?i koraci.



Slika B.5.1 Tri dodatne SSL poruke autentifikuju klijenta

Razmena poruka pri autentifikaciji klijenta	
Poruka	**Akcija**
1	Klijent šalje *ClientHello* poruku sa predlozima za izbor SSL parametara
2	Server odgovara *ServerHello* porukom sa SSL parametrima koje je izabrao
3	Server šalje *Certificate* poruku koja sadrži sertifikat servera
4	Server šalje *CertificateRequest* poruku kojom traži da autentifikuje klijenta
5	Server zaklju?uje svoj deo pregovora *ServerHelloDone* porukom
6	Klijent šalje *Certificate* poruku koja sadrži sertifikat klijenta
7	Klijent šalje klju? koji ?e se koristiti tokom sesije (kriptovan javnim klju?em servera) u *ClientKeyExchange* poruci
8	Klijent šalje *CertificateVerify* poruku koja sadrži važne informacije o sesiji potpisane privatnim klju?em klijenta; server koristi javni klju? klijenta iz sertifikata klijenta, kako bi mogao da potvrdi identitet klijenta
9	Klijent šalje *ChangeCipherSpec* poruku kojom aktivira koriš?enje zašti?ene komunikacije sa dogovorenim parametrima za sve budu?e poruke
10	Klijent šalje *Finished* poruku kako bi server proverio novoaktivirane sigurnosne opcije
11	Server šalje *ChangeCipherSpec* poruku kojom aktivira koriš?enje zašti?ene komunikacije sa dogovorenim parametrima za sve budu?e poruke
12	Server šalje *Finished* poruku kako bi klijent proverio novoaktivirane sigurnosne opcije

Tabela B.5.1 Razmena SSL poruka pri autentifikaciji klijenta

Samo slanje *Certificate* poruke od strane klijenta ne obezbe?uje potpunu autentifikaciju klijenta. Klijent mora da dokaže da poseduje i odgovaraju?i privatni klju?. Zato, klijent šalje *CertificateVerify* poruku koja sadrži digitalno potpisan heš dobijen od informacije koja je poznata i klijentu i sreveru. Na taj na?in server može da proveri potpis i utvrdi da li klijent poseduje odgovaraju?i privatni klju?.

Podrazumeva se da je server dužan da proveri da li može da veruje sertifikatu koji dobije od klijenta. To uklju?uje proveru lanca poverenja i provore validnosti sertifikata.

U navedenom primeru SSL klijent je *web* pretraživa?, a digitalni sertifikat koji mu je potreban da bi opisani scenario funkcionisao je li?ni korisni?ki sertifikat.

Prednosti korištenja PKI za autentifikaciju klijenta (krajnjih korisnika) su:

- Osetljive informacije (*password* ili privatni klju?) ne prenose se kroz mrežu.
- Nije potreban repozitorijum za korisni?ke akreditiv (*credential*) – postoji CA kome svi veruju.
- Autentifikacija nije centralizovan servis.

B.6 Kreiranje i format SSL poruke

SSL protokol se oslanja na TCP protokol preko svog *Record* protokola. *Record* protokol preuzima poruke aplikacije i poruke jednog od tri SSL *Handshake* pod-protokola (slika B.6.1), formatira ih, vrši odgovaraju?u enkapsulaciju i prosle?uje ih transportnom sloju



Slika B.6.1 Komponente SSL protokola

B.6.1 SSL i transportni sloj

Pošto zahteva puzdan prenos, bez grešaka, SSL koristi TCP na transportnom sloju. Omogu?eno je kombinovanje više SSL poruka u okviru jednog TCP segmenta.



Slika B.6.1.1 Kombinovanje više SSL poruka u jedan TCP segmentu

B.6.2 Record sloj SSL protokola

Record sloj SSL protokola vrši enkapsulaciju u format prepoznatljiv za sve SSL poruke (*Alert*, *ChangeCipherSpec*, *Handshake* i poruke aplikacije). Na slici B.6.2.1 je prikazan format poruke, a u tabeli B.6.2.1 su navedena objašnjenja za odgovaraju?a imena polja.



Slika B.6.2.1 Record sloj SSL protokola je zadužen za enkapsulaciju poruke

Polje	Veli?ina (B)	Namena
Protocol	1	Ozna?ava da je poruka deo jednog od tri *handshake* pod-protokola ili aplikacije. Mogu?e su slede?e vrednosti: 20 – *ChangeCipherSpec*; 21 – *Alert*; 22 – *Handshake*; 23 - Aplikacija
Version	2	Verzija SSL protokola. Trenutna verzija je 3.0, a TLS koristi verziju 3.1
Length	2	Dužina poruke preuzete od protokola na sloju iznad Record sloja SSLa.
Protocol Messages	n	Poruka

Tabela B.6.2.1 Polja SSL Record Layer poruke

***ChangeCipherSpec* protokol:** je jednostavan protokol koji ima samo jednu poruku, *ChangeCipherSpec*, koja je ve? pomenuta i objašnjena.

***Alert* protokol:** se koristi da bi signalizirali grešku ili upozorenje drugoj strani uklju?enoj u komunikaciju.

***Handshake* protokol:** je zadužen za razmenu poruka tokom uspostavljanja i dogovaranja o parametrima SSL sesije. Pripadaju mu poruke *HelloRequest*, *ClientHello*, *ServerHello*, *Certificate*, *ServerKeyExchange*, *CertificateRequest*, *ServerHelloDone*, *CertificateVerify*, *ClientKeyExchange*, *Finished*, ?ije je zna?enje opisano na po?etku u poglavlju B.1 SSL poruke.

From:
http://www.bpd.amres.ac.rs/ - **AMRES wiki**

Permanent link:
http://www.bpd.amres.ac.rs/doku.php?id=amres_cbp_wiki:bpd_106_dodatakb_ssl

Last update: **2011/05/15 20:49**