Transport Layer Security

Νικος Μαυρογιαννόπουλος

25 Μαρτίου 2001

1 Εισαγωγή

Αν και σήμερα οι συναλλαγές μέσω του δικτύου είναι πραγματικότητα, στις αρχές της δεκαετίας του '90 δεν υπήρχαν ουσιαστικά συναλλαγές μέσω δικτύου. Ενας απο τους λόγους (χωρίς να είναι ο μοναδικός), ήταν η ελλειψη ασφάλειας στις συναλλαγές. Δηλαδή δεν μπορούσε ο υποψήφιος πελάτης να βεβαιωθεί ότι τα στοιχεία που έδωσε (συγκεκριμένα τον αριθμο της πιστωτικής του κάρτας) τα είδε μόνο ο πωλητής.

Αυτό οδήγησε την Netscape, την εταιρία που τότε κατείχε το μεγαλύτερο μέρος της αγοράς στους Web browsers, να δημιουργήσει ένα πρωτόκολλο για ασφαλή επικοινωνία μετάξύ Web servers και browsers. Η πρώτη της προσπάθεια ήταν το SSL 2.0 το 1994. Ενα χρόνο αργότερα η Microsoft δημιούργησε τη δική της έκδοση του πρωτοκόλλου, ονομαζόμενη PCT 1.0.

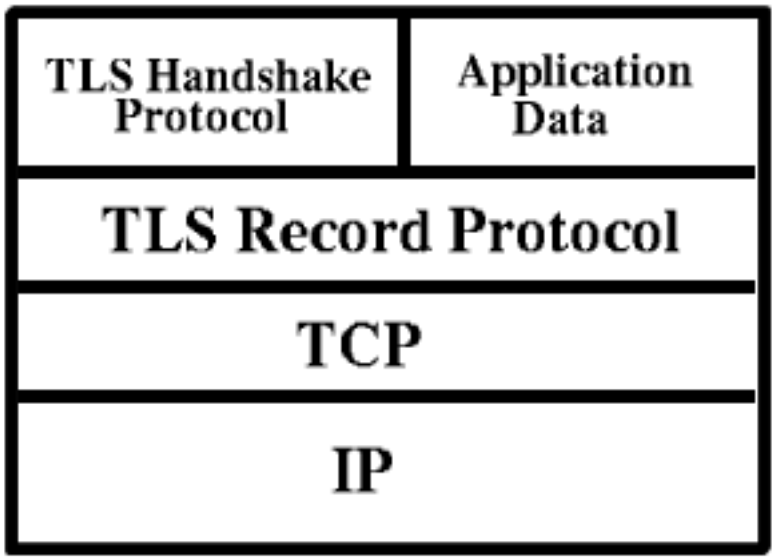
Το 1996 συστάθηκε το TLS working group από το IETF, για να τυποποιήσει ένα πρωτόκολλο ασφαλείας στο επιπεδο μεταφοράς (transport layer). Το working group ξεκίνησε με το SSL 3.0 το 1996, ενσωματώνοντας στοιχεία απο το SSL 2.0 και PCT 1.0. Το 1999 εκδόθηκε απο το TLS working group το RFC2246, που περιέγραφε το TLS 1.0 πρωτόκολλο. Αυτό ήταν ο διαδοχος του SSL 3.0 και η χρήση του σήμερα δεν περιορίζεται μόνο σε web browsers.

2 Wireless Networks

Το TLS μπορεί να χρησιμοποιηθεί και στους WAP browsers, αλλά είναι αρκετά βαρύ για να χρησιμοποιηθεί με περιορισμένο hardware και bandwidth. Εχουν προταθεί αλλαγές στο TLS πρωτόκολλο με το internet draft: Wireless Extensions to TLS, οι οποίες κάνουν το πρωτόκολλο πιο ευχρηστο σε τέτοια δίκτυα.

Ανεξάρτητα απο τις παραπάνω αλλαγές στο πρωτόκολλο, δημιουργήθηκε ένα νέο πρωτόκολλο ειδικά για ασύρματα δίκτυα, το WTLS. Το WTLS είναι βασισμένο στο TLS, αλλά σχεδιασμένο ειδικά για τους περιορισμούς των ασυρμάτων δικτύων. Περισσότερες πληροφορίες γι΄ αυτο θα βρείτε στο http://www.wapforum.org.

3 Layers



Το TLS πρωτόκολλο είναι πάνω απο το TCP/IP επίπεδο. Αυτό έχει το πλεονέκτημα ότι δεν χρειάζονται ειδικές αλλαγές στον εξοπλισμό (routers, κλπ), όπως επίσης και στα λειτουργικά συστήματα. Το μειονέκτημα είναι ότι οι εφαρμογές πρέπει να το υποστηρίζουν (δεν είναι διάφανο).

4 Τι παρέχει το TLS 1.0;

1. προστασία απο παθητικούς παρατηρητές
2. προστασια απο αλλαγή των δεδομένων
3. πιστοποίηση ταυτότητας των συνομιλουντων μερών

4.1 Προστασια απο παθητικούς παρατηρητές

Επιτυγχάνεται με την (συμμετρική) κρυπτογράφιση των δεδομένων χρησιμοποιώντας block αλγόριθμους κρυπτογράφισης σε κατάσταση CBC:

- DES
- TripleDES
- RC2
- RIJNDAEL (AES)
- RC4 (stream)

Τα κλειδιά αυτων των αλγορίθμων κυμαίνονται απο: 40bit μέχρι 256bit στον RIJNDAEL. Τα κλειδιά αυτά είναι μοναδικά ανα σύνδεση και παραγονται χρησιμοποιώντας αλγοριθμους ανταλαγής κλειδιών. Ειναι διαφορετικά ανα σύνδεση ώστε ποτέ απο τα ίδια δεδομένα να μην προκύψουν ίδια κρυπτογραφιμένα δεδομένα.

Ειδικά στους μπλοκ αλγοριθμους (3DES, RIJNDAEL) υπάρχει και προστασία απο στατιστική ανάλυση των δεδομένων. Ενας τυχαίος αριθμός μπλοκ προστίθεται στο τέλος κάθε πακέτου που ανταλλάσεται μεταξύ εξυπηρέτη-εξυπηρετούμενου.

Σε ορισμένες υλοποιήσεις παρέχεται και συμπίεση δεδομένων (αλλά έχει αρκετά μεγάλο κόστος στην ταχύτητα και στην κατανάλωση μνήμης). Η συμπίεση προβλέπεται απο το πρωτόκολλο, αλλά χωρίς να καθορίζονται συγκεκριμένοι αλγόριθμοι.

Για όλους τους παραπάνω αλγορίθμους χρειάζονται τυχαίοι (απροβλεπτοι) αριθμοι. Σε πολλά συστημάτα χρησιμοποιούνται ειδικά αρχεία του συστήματος όπως το /dev/urandom και /dev/random (στο linux).

4.2 Προστασια απο αλλαγή των δεδομένων

Αυτό επιτυγχάνεται χρησιμοποιώντας Message Authentication Code (MAC) αλγοριθμους.

Οι αλγόριθμοι αυτοί παράγουν ένα hash μιας φοράς απο ένα μήνυμα χρησιμοποιωντας ένα κρυφό κλειδι. Είναι δύσκολο χωρίς τη γνώση του κλειδιού να διαπιστωθεί η αλλαγή του μηνύματος. Χρησιμοποιούνται για να διαπιστώθει κατα πόσο έχει αλλοιωθεί το μηνύμα.

- Στο TLS 1.0 χρησιμοποιείται η HMAC κατασκευή με τους MD5 και SHA1 αλγορίθμους.

- Το SSL 3.0 χρησιμοποιεί μια MAC κατασκευή που μοιάζει με την HMAC

Τα κλειδιά που χρησιμοποιούν παράγονται απο τους αλγόριθμους ανταλαγής κλειδιών.

4.3 πιστοποίηση ταυτότητας των συνομιλουντων μερών

4.3.1 Αλγόριθμοι ανταλλαγής κλειδιών

Στο TLS χρησιμοποιούνται οι Diffie Hellman, RSA, SRP (internet draft)

- Diffie Hellman, RSA

  είναι αλγοριθμοι δημόσιου κλειδιού (public key) και επιτυγχάνουν την ανταλλαγή κλειδιών μεταξύ εξυπηρέτη-εξυπηρετούμενου

  Η ανταλλαγή κλειδιών δεν είναι γενικά ασφαλής, λόγω της man-in-the-middle επίθεσης. Χρειάζεται τρόπο πιστοποίησης.

  Οι "export" TLS υλοποιήσεις χρησιμοποιούν δημόσια κλειδιά μέχρι 512 bits.

- SRP

  είναι αλγόριθμος για την ανταλλαγή κλειδιών, όταν είναι γνωστό κάποιο συνθηματικό (password) και στα δύο μέρη

  Η υποστηριξή του στο TLS είναι ακόμη σε προκατακτικό στάδιο. Δεν υπάρχει TLS υλοποίηση που να τον υποστηρίζει.

  Πλεονεκτήματα: Δεν χρειάζεται τρόπο πιστοποίησης. Αρκεί η γνώση του συνθηματικού.

4.3.2 Public Key infrastructure

- x509 Public Key infrastructure

  Χρησιμοποιεί άλγορίθμους όπως: RSA, DSS (Digital Signature Standard) (ο RSA μπορεί να χρησιμοποιηθεί και για ψηφιακές υπογραφές και για ανταλλαγή κλειδιών)

  Χρησιμοποιείται για την αποφυγή της man in the middle επίθεσης. Παρέχει ψηφιακη πιστοποίηση του εξυπηρέτη ή και του εξυπηρετούμενου. Είναι ο βασικός τρόπος πιστοποίησης του TLS στο internet σήμερα.

  Χρησιμοποιούνται κεντρικές Certificate Authorities (CA) τις οποίες πρέπει να εμπιστευονται τα συναλλασόμενα μέρη. Οι CA παραγουν ψηφιακές υπογραφές που πιστοποιούν τα στοιχεία των συναλλασόμενων μερών.

- OpenPGP Public Key infrastructure

  Χρησιμοποιεί τους ίδιους αλγορίθμους όπως το x509 PKI, για τον ίδιο σκοπό (πιστοποίηση ταυτότητας).

Η διαφορά είναι ότι αντί να χρησιμοποιείται η έννοια της έμπιστης αρχής πιστοποίησης (CA), ο κάθε χρήστης μπορεί να πιστοποιήσει, υπογράψει τα στοιχειά οποιουδήποτε άλλου.

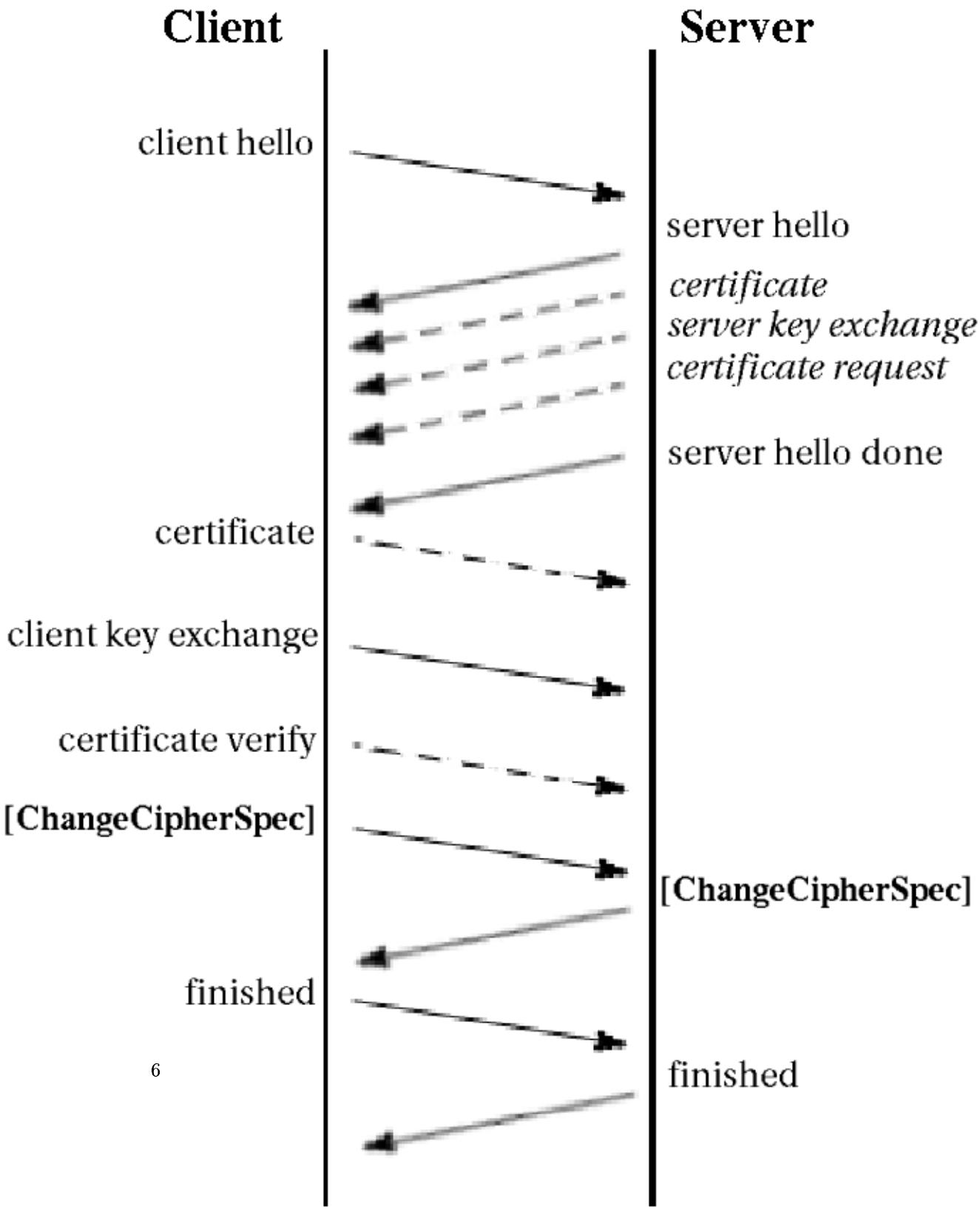
Αν και επικρατεί στο secure mail, δεν χρησιμοποιείται με το TLS σήμερα. Εχει προταθεί η χρήση του με ένα internet draft.

5 TLS Layers

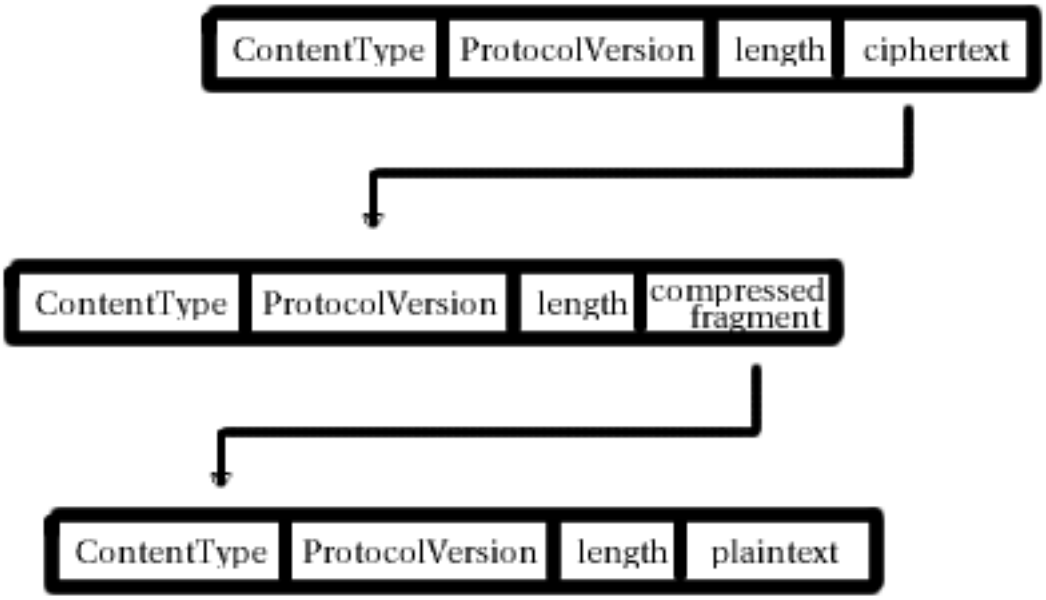
Το TLS χωρίζεται στα εξής πρωτόκολλα:

- Record Protocol: Υπευθυνο για την μεταφορά των δεδομένων
- Handshake Protocol: Υπευθυνο για την ασφαλή ανταλλαγή κλειδιών και επιβεβαίωση της ταυτότητας των συνομιλούντων μερών

5.1 TLS Handshake protocol



5.2 TLS Record protocol



6 Υλοποιήσεις του πρωτοκόλλου

- OpenSSL: Η πιο γνωστή σε παραγωγούς λογισμικού είναι η OpenSSL (http://www.openssl.org) και χρησιμοποιείται πολλές εφαρμογές (με κυριότερη τον apache).
- Network Security Services: Υπάρχει και η υλοποίηση της Netscape στο Mozilla. Περισσότερες πληροφορίες γι΄ αυτή στο http://www.mozilla.org.
- GNUTLS: Είναι σε alpha στάδιο. Περισσότερες πληροφορίες γι΄ αυτή στο http://gnutls.hellug.gr.

6.1 Απόδοση των υλοποιήσεων

Η ταχύτητα των υλοποιήσεων είναι ένα απο τα προβλήματα στους μεγάλους εξυπηρέτες.

- Σε εξυπηρέτες με μεγάλη κίνηση μπορεί να χρησιμοποιηθεί και εξειδικευμένο hardware για τους αλγόριθμους κρυπτογραφισης. Η ταχύτητα αυξάνεται σημαντικά.
- Η συμπίεση που προέβλεπε το πρωτόκολλο σπάνια χρησιμοποιείται λόγω της μεγάλης κατανάλωσης πόρων του εξυπηρέτη. (οι αλγόριθμοι συμπίεσης είναι συνήθως πιο αργοι απο τους αλγόριθμους κρυπτογράφισης και χρησιμοποιούν πολυ περισσότερη μνήμη).

6.2 Προβλήματα στα πρωτόκολλα

Τα πρωτόκολλα SSL 2.0 και PCT 1.0, αν και άνοιξαν το δρόμο στις ασφαλείς συναλλαγές μέσω internet, έχουν σοβαρά προβλήματα ασφαλείας και δεν πρέπει να χρησιμοποιούνται σήμερα.