Εργαστηριακή Άσκηση 12
Ασφάλεια

Στα δίκτυα επικοινωνίας ένας εισβολέας μπορεί: να υποκλέπτει μηνύματα κρυφακούγοντας τις μεταδόσεις, να εισάγει αυθαίρετα μηνύματα στη σύνδεση, να υποδύεται άλλον παράγοντας μηνύματα με ψευδή διεύθυνση πηγής καθώς και να αποστερεί τη χρήση υπηρεσιών από άλλους (πχ. υπερφορτώνοντας το δίκτυο). Τα βασικά θέματα ασφαλείας, όσον αφορά την επικοινωνία πάνω από δίκτυα δεδομένων, περιλαμβάνουν την πιστοποίηση αυθεντικότητας (authentication), τον έλεγχο πρόσβασης (access control), την εμπιστευτικότητα (confidentiality), την ακεραιότητα (integrity), τη μη αποποίηση ευθύνης (non-repudiation), τη διαθεσιμότητα (availability) και το απόρρητο (privacy). Μέσω της πιστοποίησης αυθεντικότητας ο αποστολέας και ο παραλήπτης επιβεβαιώνουν ο ένας την ταυτότητα του άλλου. Μέσω του ελέγχου πρόσβασης οι υπηρεσίες είναι προσβάσιμες μόνο στους εξουσιοδοτημένους χρήστες. Μόνο ο αποστολέας και ο "κανονικός" παραλήπτης πρέπει να κατανοούν το περιεχόμενο του μηνύματος (εμπιστευτικότητα): ο αποστολέας κρυπτογραφεί το μήνυμα και ο παραλήπτης το αποκρυπτογραφεί. Επιπλέον, ο αποστολέας και ο παραλήπτης θέλουν να είναι βέβαιοι ότι το μήνυμα δεν τροποποιήθηκε κατά τη διαδρομή (ή μεταγενέστερα) χωρίς αυτό να γίνει αντιληπτό (ακεραιότητα). Ο χρήστης της υπηρεσίας δε θα πρέπει να μπορεί να αρνηθεί τη χρήση της (μη αποποίηση ευθύνης) και ένα ελάχιστο επίπεδο υπηρεσίας πρέπει να είναι διαθέσιμο (διαθεσιμότητα). Τέλος το απόρρητο (privacy) περιλαμβάνει τα στοιχεία επικοινωνίας που πρέπει να είναι μυστικά στους τρίτους, δηλαδή, με ποιον επικοινωνεί ο χρήστης, σε ποια θέση, κλπ.

Ο σκοπός αυτής της εργαστηριακής άσκησης είναι η εξέταση θεμάτων σχετικών με την ασφαλή μετάδοση δεδομένων στο διαδίκτυο. Όπως και στις προηγούμενες εργαστηριακές ασκήσεις θα εργασθείτε με τον αναλυτή πρωτοκόλλων Wireshark. Εδώ θα χρησιμοποιήσετε τη λειτουργία *Capture,* με φίλτρο, ώστε τα πλαίσια που καταγράφονται να περιέχουν κάποια συγκεκριμένα χαρακτηριστικά. Υπενθυμίζεται ότι το φίλτρο απεικόνισης *(Display)* που επιλέγετε από το μενού *Analyze* ενεργοποιείται **αφού** έχει ολοκληρωθεί η διαδικασία καταγραφής, ώστε να αποκρύψει κάποια από τα συλληφθέντα πλαίσια, ενώ το φίλτρο σύλληψης που επιλέγετε από το μενού *Capture* ενεργοποιείται **πριν** ξεκινήσει η διαδικασία καταγραφής, με αποτέλεσμα να καταγράφεται μόνο ένα μέρος των διερχόμενων πλαισίων.

1. Πιστοποίηση αυθεντικότητας στο πρωτόκολλο HTTP

Κάποιες ιστοσελίδες στο διαδίκτυο προστατεύονται από συνθηματικά (password-protected) και δεν επιτρέπεται σε όλους τους χρήστες να τις επισκεφτούν παρά μόνο σε όσους διαθέτουν το συνθηματικό. Η διαδικασία της πιστοποίησης αυθεντικότητας (authentication) συνίσταται στην επαλήθευση της ψηφιακής ταυτότητας του χρήστη που αιτείται για είσοδο στην ιστοθέση.

Με τη βοήθεια του Wireshark, καταγράψτε την κίνηση ενώ κάνετε χρήση της υπηρεσίας HTTP του υπολογιστή `edu-dy.cn.ntua.gr` (`147.102.40.9`). Επειδή τα μηνύματα HTTP μεταφέρονται σε περισσότερα από ένα πακέτα, αφού ξεκινήσετε το Wireshark, ακολουθήστε από το μενού του κεντρικού παραθύρου τη διαδρομή *Edit* → *Preferences…*, κάνετε κλικ στο σύμβολο δίπλα στο Protocols στην αριστερή πλευρά του παραθύρου, κατόπιν εντοπίστε και κάνετε κλικ στο πρωτόκολλο HTTP και βεβαιωθείτε ότι όλες οι επιλογές περί ανασύνθεσης και αποσυμπίεσης είναι επιλεγμένες. Στη συνέχεια, στο πρωτόκολλο TCP βεβαιωθείτε ότι το *Allow subdissector to reassemble TCP streams* είναι επιλεγμένο και φροντίστε το *Validate the TCP checksum if possible*[1] να **μην** είναι επιλεγμένο. Τέλος, πιέστε OK για να κλείσει το παράθυρο και να εφαρμοσθούν οι

[1] Η επιβεβαίωση του TCP checksum (επειδή αυτή γίνεται στην κάρτα δικτύου) παρενοχλεί τη διαδικασία επανασύνθεσης.

αλλαγές σας. Με τις επιλογές αυτές, μηνύματα που μεταφέρονται σε περισσότερα από ένα πακέτα θα αποκωδικοποιηθούν από το Wireshark ως πλήρη μηνύματα HTTP και όχι αποσπασματικά.

Εφαρμόστε φίλτρο σύλληψης `host 147.102.40.9` για να παρατηρείτε μόνο την κίνηση που σχετίζεται με το `edu-dy.cn.ntua.gr`. Ξεκινήστε μία καταγραφή κίνησης και επισκεφτείτε τη σελίδα http://edu-dy.cn.ntua.gr/auth/. Η πρόσβαση σε αυτή τη σελίδα απαιτεί την επαλήθευση της ταυτότητάς σας. Δώστε `edu-dy` στο πεδίο του ονόματος χρήστη (user name) και `password` στο πεδίο του μυστικού κωδικού (password). Σταματήστε την καταγραφή.

1.1 Να καταγραφεί ο αριθμητικός κωδικός κατάστασης (status code) και η φράση που επιστρέφει ο εξυπηρετητής ως απόκριση στο αρχικό μήνυμα HTTP τύπου GET του πλοηγού ιστού.

1.2 Ο πλοηγός ιστού στέλνει και δεύτερο μήνυμα HTTP τύπου GET στον εξυπηρετητή. Συγκρίνοντας με το πρώτο μήνυμα HTTP GET να καταγραφεί το επιπλέον πεδίο που περιλαμβάνει η επικεφαλίδα HTTP του δεύτερου μηνύματος.

1.3 Καταγράψτε το περιεχόμενο του παραπάνω πεδίου όπως αυτό εμφανίζεται στο παράθυρο με τα περιεχόμενα του επιλεγμένου πλαισίου σε μορφή ASCII.

Τα στοιχεία πιστοποίησης αυθεντικότητας που καταχωρήσατε δεν κρυπτογραφήθηκαν για να αποσταλούν στον εξυπηρετητή, απλώς κωδικοποιήθηκαν σύμφωνα με μια πολύ γνωστή μέθοδο, τη Base64. Το αποτέλεσμα της κωδικοποίησης είναι η γραμματοσειρά που ακολουθεί τις λέξεις "Authorization: Basic" στην επικεφαλίδα του δεύτερου μηνύματος HTTP GET του πλοηγού.

1.4 Επισκεφτείτε την ιστοσελίδα http://www.motobit.com/util/base64-decoder-encoder.asp και εισάγετε στο παράθυρο που εμφανίζεται το περιεχόμενο του πεδίου που καταγράψατε στο ερώτημα 1.3. Αποκωδικοποιήστε το περιεχόμενο αυτό επιλέγοντας το "**decode** the data **from a Base64** string (base64 decoding)" και κάνοντας κλικ στο κουμπί "Convert the source data". Καταγράψτε το αποτέλεσμα της αποκωδικοποίησης[2].

1.5 Τι συμπεραίνετε για την ασφάλεια του βασικού μηχανισμού πιστοποίησης αυθεντικότητας που παρέχει το HTTP; *[Υπόδειξη: Συμβουλευτείτε την παράγραφο "Basic Authentication Scheme" από το RFC 1945: Hypertext Transfer Protocol – HTTP/1.0]*

2. Υπηρεσία SSH – Secure SHell

Το Secure Shell ή SSH είναι ένα πρωτόκολλο που επιτρέπει την ανταλλαγή δεδομένων μεταξύ δύο δικτυακών οντοτήτων μέσω ενός ασφαλούς διαύλου επικοινωνίας. Σε λειτουργικά συστήματα τύπου Unix χρησιμοποιείται αντί του TELNET ή άλλων μη ασφαλών προγραμμάτων πρόσβασης στον φλοιό (rlogin, κλπ), τα οποία στέλνουν πληροφορία, όπως π.χ. τους κωδικούς χρήστη, χωρίς κρυπτογράφηση. Συνήθως χρησιμοποιείται για εκτέλεση εντολών στον απομακρυσμένο υπολογιστή. Υποστηρίζει όμως σήραγγες, συνδέσεις X11 καθώς και μεταφορά αρχείων με τη βοήθεια των συνδεδεμένων πρωτοκόλλων SFTP ή SCP. Περισσότερες πληροφορίες για το πρωτόκολλο SSH θα βρείτε στο http://en.wikipedia.org/wiki/Secure_Shell και στο RFC 4251: The Secure Shell (SSH) Protocol Architecture. Η κρυπτογράφηση που χρησιμοποιείται στο SSH παρέχει εμπιστευτικότητα και διασφαλίζει την ακεραιότητα (integrity) των μεταδιδόμενων δεδομένων. Το SSH χρησιμοποιεί κρυπτογράφηση δημόσιου κλειδιού για την πιστοποίηση αυθεντικότητας του απομακρυσμένου υπολογιστή και επιτρέπει την πιστοποίηση αυθεντικότητας του τοπικού υπολογιστή από τον απομακρυσμένο υπολογιστή.

Μια σύντομη περιγραφή της λειτουργίας του πρωτοκόλλου SSH ακολουθεί. Ο πελάτης και εξυπηρετητής SSH διαπραγματεύονται την εγκατάσταση σύνδεσης ακολουθώντας μια διαδικασία διαπραγμάτευσης. Η σύνδεση ξεκινά από την πλευρά του πελάτη. Αμφότερες οι πλευρές πρέπει να στείλουν ένα αναγνωριστικό κείμενο (string) της μορφής "SSH-protoversion-softwareversion comments", το οποίο περιλαμβάνει *υποχρεωτικά* τις εκδόσεις του πρωτοκόλλου SSH (protoversion) και του αντίστοιχου λογισμικού (softwareversion) και *προαιρετικά* κάποια σχόλια (comments).

[2] Το Wireshark αυτομάτως εκτελεί την αποκωδικοποίηση αυτή. Μπορείτε να δείτε το αποτέλεσμα κάνοντας διπλό κλικ στο πεδίο "Authorization: Basic" της επικεφαλίδας HTTP.

Ακολουθεί η διαπραγμάτευση για τους αλγόριθμους κρυπτογράφησης (encryption) δεδομένων, κατακερματισμού (hash) για την παραγωγή κωδικών πιστοποίησης αυθεντικότητας μηνυμάτων (MAC – Message Authentication Code) και συμπίεσης (compression) δεδομένων. Οι ροές δεδομένων κατά τις κατευθύνσεις πελάτης→εξυπηρετητής, εξυπηρετητής→πελάτης είναι ανεξάρτητες και επιτρέπεται να χρησιμοποιηθούν διαφορετικοί αλγόριθμοι (π.χ 3DES+SHA1 και Blowfish+MD5). Εάν χρησιμοποιηθεί συμπίεση, τότε τα δεδομένα συμπιέζονται πρώτα και κατόπιν κρυπτογραφούνται. Στη συνέχεια εγκαθίστανται τα κλειδιά κρυπτογράφησης και ακεραιότητας. Κάθε μήνυμα που ακολουθεί κρυπτογραφείται με το κλειδί κρυπτογράφησης και η γνησιότητά του πιστοποιείται με την προσθήκη του κωδικού MAC που παράγεται από τα περιεχόμενα του μηνύματος και τον αύξοντα αριθμό του βάση της συνάρτησης κατακερματισμού και του αντίστοιχου κλειδιού ακεραιότητας.

ΠΡΟΣΟΧΗ: Για τη χρήση της υπηρεσίας SSH θα χρησιμοποιήσετε μια παλιότερη[3] έκδοση του προγράμματος PuTTY αντί αυτής που βρίσκεται στην επιφάνεια εργασίας (`putty.exe`). Προκειμένου να κατεβάσετε τη συγκεκριμένη έκδοση του PuTTY θα χρησιμοποιήσετε την υπηρεσία FTP του υπολογιστή `edu-dy.cn.ntua.gr`. Ανοίξτε νέο παράθυρο εντολών και πληκτρολογήστε `ftp edu-dy.cn.ntua.gr`. Στην προτροπή User: πληκτρολογήστε `anonymous` ακολουθούμενο από `<Enter>`, ενώ στην προτροπή Password: πληκτρολογήστε το e-mail σας ακολουθούμενο από `<Enter>`. Στη συνέχεια, πληκτρολογήστε την εντολή `bin`, ώστε η μεταφορά αρχείων να γίνει σε δυαδική μορφή. Επιλέξτε την επιφάνεια εργασίας ως τον προορισμό όπου επιθυμείτε να αποθηκεύσετε το πρόγραμμα με την εντολή `lcd desktop`. Κατεβάστε το `putty-058.exe` με την εντολή `get putty-058.exe`. Τέλος, πληκτρολογήσετε `bye` για να τερματίσετε την εφαρμογή `ftp`.

Αφού βεβαιωθείτε ότι το πρόγραμμα `putty-058.exe` βρίσκεται πλέον στην επιφάνεια εργασίας του υπολογιστή σας, καταγράψτε με τη βοήθεια του Wireshark την κίνηση ενώ κάνετε χρήση της υπηρεσίας SSH του υπολογιστή `edu-dy.cn.ntua.gr`. Όπως πριν, εφαρμόστε φίλτρο σύλληψης `host 147.102.40.9` για να παρατηρείτε μόνο την κίνηση που σχετίζεται με το `edu-dy.cn.ntua.gr` και ξεκινήστε την καταγραφή. Στο πεδίο *Host Name* του παραθύρου PuTTY που ανοίγει όταν εκτελέσετε το `putty-058.exe`, πληκτρολογήστε `edu-dy.cn.ntua.gr`, στη συνέχεια κάνετε κλικ στο πρωτόκολλο *SSH* και τέλος στο κουμπί *Open*. Αν ενδεχομένως ανοίξει κάποιο παράθυρο διαλόγου, επιλέξατε *Yes* για να προχωρήσετε. Στην προτροπή `login:` πληκτρολογήστε `abcd` ακολουθούμενο από `<Enter>`, ενώ στην προτροπή `Password:` πληκτρολογήστε `efgh` ακολουθούμενο από `<Enter>`. Σημειώνεται ότι ο χρήστης abcd δεν υπάρχει στον συγκεκριμένο εξυπηρετητή και η αναγνώριση του χρήστη θα αποτύχει. Πληκτρολογήστε `<Ctrl>+c` για να κλείσει το παράθυρο και σταματήσετε την καταγραφή κίνησης.

2.1 Ποιο πρωτόκολλο μεταφοράς χρησιμοποιεί το SSH (TCP ή UDP);

2.2 Καταγράψτε τις θύρες του πρωτοκόλλου μεταφοράς που χρησιμοποιούνται για την επικοινωνία μεταξύ του υπολογιστή σας και του `edu-dy.cn.ntua.gr`.

2.3 Ποια από τις παραπάνω θύρες αντιστοιχεί στο πρωτόκολλο εφαρμογής SSH; *[Προσδιορίστε τη ζητούμενη θύρα συμβουλευόμενοι τον κατάλογο πασίγνωστων θυρών στην ιστοσελίδα http://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers.]*

2.4 Αναλύοντας το αναγνωριστικό που στέλνει ο εξυπηρετητής στον πελάτη, ποια έκδοση του πρωτοκόλλου SSH και ποια έκδοση λογισμικού χρησιμοποιεί ο εξυπηρετητής; Περιλαμβάνονται σχόλια στο αναγνωριστικό αυτό; Αν ναι, να καταγραφούν.

[3] Το Wireshark παρουσιάζει προβλήματα κατά την αποκωδικοποίηση ορισμένων μηνυμάτων SSH που παράγουν οι νεότερες εκδόσεις του PuTTY.

2.5 Αναλύοντας το αναγνωριστικό που στέλνει ο πελάτης στον εξυπηρετητή, ποια έκδοση του πρωτοκόλλου SSH και ποια έκδοση λογισμικού χρησιμοποιεί ο πελάτης; Περιλαμβάνονται σχόλια στο αναγνωριστικό αυτό; Αν ναι, να καταγραφούν.

2.6 Μπορείτε να εντοπίσετε τα πακέτα όπου μεταφέρεται η πληροφορία για την προτροπή login και password στην περίπτωση του SSH; Να δικαιολογήσετε την απάντησή σας.

2.7 Εντοπίσετε τη λίστα με τους αλγόριθμους κρυπτογράφησης (encryption algorithms) που υποστηρίζει ο πελάτης, αναζητώντας μέσα στο μήνυμα SSH τύπου *Key Exchange Init* και καταγράψτε τους δύο πρώτους για την κατεύθυνση πελάτης → εξυπηρετητής.

2.8 Εντοπίσετε τη λίστα με τις συναρτήσεις κατακερματισμού (mac algorithms) που υποστηρίζει ο πελάτης και καταγράψτε τους δύο πρώτους για την κατεύθυνση εξυπηρετητής → πελάτης.

2.9 Αφού εντοπίσετε τη λίστα με τους αλγόριθμους κρυπτογράφησης που υποστηρίζει ο εξυπηρετητής, βρείτε αυτόν που τελικά θα χρησιμοποιηθεί στην κατεύθυνση πελάτης → εξυπηρετητής. *[Υπόδειξη: Όπως μπορείτε να βρείτε στο RFC 4253: SSH Transport Layer Protocol, αναζητώντας τον όρο "encryption_algorithms", ο αλγόριθμος κρυπτογράφησης είναι ο πρώτος της λίστας του πελάτη που υπάρχει και στην λίστα του εξυπηρετητή]*.

2.10 Βρείτε τη συνάρτηση κατακερματισμού που τελικά θα χρησιμοποιηθεί στην κατεύθυνση εξυπηρετητής → πελάτης.

2.11 Σχολιάστε την ασφάλεια της υπηρεσίας SSH όσον αφορά την πιστοποίηση της αυθεντικότητας, την εμπιστευτικότητα και την ακεραιότητα των δεδομένων συγκρίνοντας με άλλα πρωτόκολλα ανταλλαγής δεδομένων.

3. Υπηρεσία HTTPS

Το Hypertext Transfer Protocol Secure (HTTPS) συνήθως χρησιμοποιείται για τη διεξαγωγή χρηματικών συναλλαγών μέσω του παγκόσμιου ιστού καθώς και την πρόσβαση σε ευαίσθητα δεδομένα (π.χ. μηνύματα ηλεκτρονικού ταχυδρομείου). Για τη χρήση του HTTPS, αντί για `http://`, στο URI των ιστοσελίδων χρησιμοποιείται το `https://`. Το HTTPS είναι ένας συνδυασμός του Hypertext Transfer Protocol (HTTP) με ένα πρωτόκολλο για ασφαλή μετάδοση δεδομένων. Αμφότερα, το HTTP και το πρωτόκολλο ασφαλούς μετάδοσης, λειτουργούν πάνω από το στρώμα μεταφοράς TCP του διαδικτύου. Το πρωτόκολλο ασφαλούς μετάδοσης λειτουργεί ως υπόστρωμα *πάνω* από το πρωτόκολλο μεταφοράς και *κάτω* από το στρώμα εφαρμογής, κρυπτογραφώντας τα μηνύματα HTTP πριν τη μετάδοση και αποκρυπτογραφώντας τα κατά τη λήψη. Το HTTPS ήταν γνωστό και ως "Hypertext Transfer Protocol over Secure Socket Layer", αλλά τώρα το πρωτόκολλο ασφαλούς μεταφοράς είναι το Transport Layer Security (TLS) αντί του Secure Sockets Layer (SSL).

Το SSL αναπτύχθηκε αρχικά από την εταιρεία Netscape το 1995 για χρήση από τους πλοηγούς ιστού κατά την κρυπτογράφηση των πληροφοριών που ανταλλάσσονται μέσω ιστού. Το TLS πρόκειται για μια βελτιωμένη έκδοση του πρωτοκόλλου SSL και συγκεκριμένα βασίστηκε στην έκδοση 3 αυτού (SSLv3). Περισσότερες πληροφορίες για τα πρωτόκολλα αυτά μπορείτε να βρείτε στην ιστοσελίδα http://www.openssl.org/related/ssl.html.

Μια σύντομη περιγραφή της λειτουργίας του πρωτοκόλλου TLS ακολουθεί. Ο πελάτης και εξυπηρετητής TLS διαπραγματεύονται την εγκατάσταση σύνδεσης ακολουθώντας μια διαδικασία χειραψίας. Κατά τη χειραψία ο πελάτης και ο εξυπηρετητής συμφωνούν σε διάφορες παραμέτρους σχετικές με την ασφάλεια της σύνδεσης. Η χειραψία αρχίζει όταν ο πελάτης ζητά μια ασφαλή σύνδεση στέλνοντας στον εξυπηρετητή ένα μήνυμα *ClientHello* και παρουσιάζοντας μια λίστα των υποστηριζόμενων κωδίκων κρυπτογράφησης (ciphers) και συναρτήσεων κατακερματισμού (hash functions). Ο εξυπηρετητής απαντά με το μήνυμα *ServerHello* και επιλέγει από τη λίστα τον κώδικα κρυπτογράφησης και τη συνάρτηση κατακερματισμού.

Κατόπιν, ο εξυπηρετητής αποστέλλει[4] στον πελάτη μέσω του μηνύματος *Certificate* την ταυτότητά του με τη μορφή ενός ψηφιακού πιστοποιητικού (digital certificate). Το πιστοποιητικό συνήθως

[4] Ανάλογα με τον επιλεχθέντα κώδικα, το βήμα αυτό μπορεί να παραληφθεί.

περιέχει ο όνομα του εξυπηρετητή, την έμπιστη αρχή πιστοποίησης (trusted certificate authority – CA) και το δημόσιο κλειδί κρυπτογράφησης του εξυπηρετητή. Ο πελάτης μπορεί να επικοινωνήσει με τον CA και να επιβεβαιώσει ότι το πιστοποιητικό είναι αυθεντικό προτού προχωρήσει στην εγκατάσταση κλειδιού κρυπτογράφησης για τη σύνοδο. Ο εξυπηρετητής αποστέλλει το μήνυμα *ServerHelloDone* υποδηλώνοντας ότι ολοκλήρωσε από την πλευρά του τη χειραψία.

Για την παραγωγή του κλειδιού συνόδου, ο πελάτης κρυπτογραφεί ένα τυχαίο αριθμό με το δημόσιο κλειδί του εξυπηρετητή και στέλνει το αποτέλεσμα στον εξυπηρετητή με το μήνυμα *ClientKeyExchange*. Μόνο ο εξυπηρετητής μπορεί να το αποκρυπτογραφήσει χρησιμοποιώντας το μυστικό του κλειδί. Με τον τρόπο αυτό ο εξυπηρετητής και ο πελάτης μοιράζονται ένα κοινό μυστικό που δεν είναι προσβάσιμο από τρίτους. Με το κοινό αυτό μυστικό και οι δύο πλευρές παράγουν το κλειδί συνόδου για την (από)κρυπτογράφηση των δεδομένων. Ο πελάτης στέλνει το μήνυμα *ChangeCipherSpec* λέγοντας στον εξυπηρετητή ότι η επικοινωνία από εδώ και πέρα θα είναι κρυπτογραφημένη. Κατόπιν, ο πελάτης στέλνει το κρυπτογραφημένο μήνυμα *EncryptedHandshakeMessage* που περιέχει το αποτέλεσμα της συνάρτησης κατακερματισμού επί των προηγούμενων μηνυμάτων της χειραψίας. Ο εξυπηρετητής θα προσπαθήσει να το αποκρυπτογραφήσει και να επιβεβαιώσει την ορθότητα του αποτελέσματος της συνάρτησης κατακερματισμού. Εάν αυτό γίνει επιτυχώς, ο εξυπηρετητής στέλνει το δικό του *ChangeCipherSpec* καθώς και το κρυπτογραφημένο μήνυμα *EncryptedHandshakeMessage*. Ο πελάτης το αποκρυπτογραφεί και επιβεβαιώνει την ορθότητα του αποτελέσματος της συνάρτησης κατακερματισμού. Εάν κάποιο από τα προηγούμενα βήματα αποτύχει, η χειραψία αποτυγχάνει και δεν εγκαθίσταται σύνδεση. Από εδώ και πέρα τα μηνύματα εφαρμογής *ApplicationData* είναι κρυπτογραφημένα. Τυχόν λάθη ή οι προειδοποιήσεις κατά τη διάρκεια της χειραψίας ή της μεταφοράς δεδομένων σηματοδοτούνται με μηνύματα *Alert*. Σε περίπτωση θανάσιμου λάθους (fatal error), η σύνοδος θα διακοπεί αμέσως. Εάν πρόκειται για προειδοποίηση (warning), η πλευρά που το λαμβάνει αποφασίζει για το κατά πόσο θα συνεχίσει τη σύνοδο.

Σε αυτή την άσκηση θα καταγραφούν τα μηνύματα που παράγονται κατά τη χρήση της υπηρεσίας HTTPS του υπολογιστή `my.ntua.gr`. Προτού αρχίσετε την καταγραφή φροντίστε να αδειάσετε την προσωρινή μνήμη (cache) του Firefox. Κατόπιν ξεκινήστε μια νέα καταγραφή εφαρμόζοντας φίλτρο σύλληψης ώστε να παρατηρείτε μόνο την κίνηση που σχετίζεται με τον `my.ntua.gr`. Πρώτα, επισκεφθείτε με τον Firefox την ιστοσελίδα `http://my.ntua.gr`. Μόλις φορτωθεί η σελίδα, επισκεφθείτε την πάλι, χρησιμοποιώντας αυτή τη φορά το πρωτόκολλο *HTTPS*. Για το σκοπό αυτό, πληκτρολογήστε τη διεύθυνση `https://my.ntua.gr` και δεχθείτε το πιστοποιητικό που επιδίδει ο εξυπηρετητής πιέζοντας *OK* στο παράθυρο διαλόγου που ενδεχομένως εμφανισθεί κατά το φόρτωμα της σελίδας. Όταν φορτωθεί πλήρως η σελίδα σταματήστε την καταγραφή. Να σημειωθεί ότι το Wireshark εμφανίζει τα πακέτα που μεταφέρουν τα μηνύματα του HTTPS ως SSL (Secure Sockets Layer) ή TLSv1 (Transport Layer Security).

3.1 Ποια είναι η σύνταξη του φίλτρου σύλληψης που χρησιμοποιήσατε;
3.2 Εφαρμόστε κατάλληλο φίλτρο απεικόνισης ώστε να παραμείνουν μόνο τα πρώτα τεμάχια TCP των τριμερών χειραψιών που διεξήχθησαν με τον εξυπηρετητή `my.ntua.gr`. Ποια είναι η σύνταξή του;
3.3 Σε ποιες (πασίγνωστες) θύρες του εξυπηρετητή `my.ntua.gr` γίνονται οι συνδέσεις;
3.4 Ποια από τις παραπάνω θύρες αντιστοιχεί στο πρωτόκολλο εφαρμογής HTTP και ποια στο HTTPS; *[Προσδιορίστε τις ζητούμενες θύρες συμβουλευόμενοι την ιστοσελίδα http://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers.]*
3.5 Βρείτε πόσες συνδέσεις ανοίχθηκαν μεταξύ του υπολογιστή σας και του εξυπηρετητή ιστού `my.ntua.gr` στην περίπτωση HTTP και πόσες στην περίπτωση HTTPS.

Σε αυτό το σημείο πρέπει να αναφερθεί ότι το πρωτόκολλο SSL/TLS αποτελείται στην πραγματικότητα από δύο στρώματα. Στο κατώτερο επίπεδο και πάνω από κάποιο αξιόπιστο πρωτόκολλο μεταφοράς (π.χ. το TCP), είναι το Στρώμα Εγγραφών (Record Layer) SSL/TLS. Το

στρώμα αυτό χρησιμοποιείται για την ενθυλάκωση κάποιου πρωτοκόλλου SSL/TLS ανώτερου επιπέδου, όπως είναι, το Πρωτόκολλο Χειραψίας (Handshake Protocol), το πρωτόκολλο συναγερμών (Alert Protocol), το πρωτόκολλο μετάβασης σε κρυπτογράφηση (ChangeCipherSpec) και το πρωτόκολλο εφαρμογής (Apllication),. Πρέπει επίσης να τονιστεί ότι κάθε πλαίσιο Ethernet μπορεί να περιλαμβάνει μία ή περισσότερες εγγραφές SSL/TLS. Επιπλέον, στην περίπτωση που μια εγγραφή SSL/TLS δε χωράει σε ένα πλαίσιο Ethernet, τότε θα χρειαστούν πολλαπλά πλαίσια για να τη μεταφέρουν.

Εφαρμόστε το φίλτρο απεικόνισης `ssl.record` ώστε να παραμείνουν μόνο πλαίσια τα οποία περιλαμβάνουν εγγραφές SSL. Υπενθυμίζεται ότι οι εγγραφές αυτές δημιουργήθηκαν από τη χρήση του πρωτοκόλλου HTTPS με τον `my.ntua.gr`.

3.6 Αναπτύσσοντας τις επικεφαλίδες Στρώματος Εγγραφών SSL/TLS κάθε πλαισίου θα παρατηρήσετε ότι τα τρία πρώτα πεδία είναι κοινά. Ποια είναι αυτά και ποιο το μήκος τους;

3.7 Ένα από τα πεδία που καταγράψατε στο παραπάνω ερώτημα είναι ο *τύπος περιεχομένου* (*content type*). Να καταγραφούν οι διαφορετικές τιμές για όλες τις εγγραφές SSL/TLS που έχετε καταγράψει (π.χ. Change Cipher Spec – 20). *[Υπόδειξη: Υπενθυμίζεται ότι ορισμένα πλαίσια μπορεί να περιλαμβάνουν περισσότερες από μία εγγραφές SSL/TLS]*

3.8 Εντοπίσατε τύπο περιεχομένου *Alert – 21* μεταξύ των τιμών που καταγράψατε στο προηγούμενο ερώτημα; Αν ναι, ποιος νομίζετε ότι ήταν ο σκοπός του; *[Υπόδειξη: Συμβουλευτείτε την ιστοσελίδα http://en.wikipedia.org/wiki/Transport_Layer_Security, παράγραφος Alert Protocol, για τις πιθανές προειδοποιήσεις που σηματοδοτεί το Alert.]*

3.9 Εντοπίστε το πρώτο μήνυμα *Client Hello* που στέλνει ο πελάτης κατά τη χειραψία του πρωτοκόλλου SSL/TLS. Ποιος είναι ο τύπος περιεχομένου της αντίστοιχης εγγραφής SSL/TLS;

3.10 Εξετάζοντας την επικεφαλίδα της παραπάνω εγγραφής SSL/TLS, αναπτύξτε τη λίστα με τις σουίτες πρωτοκόλλων κρυπτογράφησης (cipher suites) που υποστηρίζει ο πελάτης. Να καταγραφεί το πλήθος τους και οι δεκαεξαδικές τιμές των δύο πρώτων από αυτές.

3.11 Εξετάζοντας την επικεφαλίδα εγγραφής SSL/TLS του μηνύματος που χρησιμοποιεί ο εξυπηρετητής για να απαντήσει στη χειραψία (*Server Hello*), καταγράψτε τη δεκαεξαδική τιμή της σουίτας πρωτοκόλλων κρυπτογράφησης η οποία τελικά επιλέχθηκε.

3.12 Εντοπίστε την εγγραφή SSL/TLS που μεταφέρει το πιστοποιητικό (*Certificate*) του εξυπηρετητή. Ποιο είναι το μήκος αυτής σύμφωνα με το πεδίο length της επικεφαλίδας και πόσα πιστοποιητικά μεταφέρονται;

3.13 Πόσα πλαίσια Ethernet χρειάστηκαν ώστε να μεταφερθεί η παραπάνω εγγραφή SSL/TLS; *[Υπόδειξη: Αναπτύξτε το πεδίο [Reassembled TCP Segments] που εμφανίζει το Wireshark στις λεπτομέρειες του πλαισίου που αντιστοιχεί στη συγκεκριμένη εγγραφή.]*

3.14 Εντοπίστε την εγγραφή SSL/TLS για την παραγωγή του κλειδιού συνόδου (*ClientKeyExchange*). Ποιο είναι το μήκος του κρυπτογραφημένου μηνύματος που μεταφέρει σύμφωνα με το πεδίο length της επικεφαλίδας;

3.15 Ποιο είναι το μήκος της εγγραφής SSL/TLS (*ChangeCipherSpec*) που υποδεικνύει στον εξυπηρετητή ότι η επικοινωνία από εδώ και πέρα θα είναι κρυπτογραφημένη;

3.16 Ποιο είναι το μήκος της εγγραφής SSL/TLS (*EncryptedHandshakeMessage*) που περιέχει το αποτέλεσμα της συνάρτησης κατακερματισμού επί των προηγούμενων μηνυμάτων της χειραψίας;

3.17 Παρατηρήσατε αποστολή εγγραφών SSL/TLS (*ChangeCipherSpec*) και (*EncryptedHandshakeMessage*) από τον εξυπηρετητή;

3.18 Αφού απενεργοποιήσετε το ισχύον φίλτρο απεικόνισης, επιλέξτε από την ιστοσελίδα μια φράση με λατινικούς χαρακτήρες (π.χ. "`Mail Server`"). Προσπαθήστε να βρείτε το πακέτο που μεταφέρει αυτή την πληροφορία. Τι παρατηρείτε στην περίπτωση του πρωτοκόλλου HTTP σε σύγκριση με αυτή του HTTPS; *[Υπόδειξη: Για την εύρεση του ζητούμενου πακέτου, ακολουθήστε τη διαδρομή Edit → Find Packet… και πληκτρολογήστε τη φράση προς αναζήτηση αφού βεβαιωθείτε ότι η επιλογή String είναι ενεργή.]*

3.19 Σχολιάστε την ασφάλεια του πρωτοκόλλου HTTPS σε σύγκριση με το απλό HTTP, όσον αφορά την πιστοποίηση της αυθεντικότητας, την εμπιστευτικότητα και την ακεραιότητα των δεδομένων.

Ονοματεπώνυμο:	**Όνομα PC:**
Ομάδα:	**Ημερομηνία:**
Διεύθυνση IP: . . .	**Διεύθυνση MAC: - - - - -**

Εργαστηριακή Άσκηση 12
Ασφάλεια

Απαντήστε στα ερωτήματα στον χώρο που σας δίνεται παρακάτω και στην πίσω σελίδα εάν δεν επαρκεί. Το φυλλάδιο αυτό θα παραδοθεί στον επιβλέποντα.

1

1.1 .....

1.2 .....

1.3 .....

.....

1.4 .....

1.5 .....

.....

.....

2

2.1 .....

2.2 .....

2.3 .....

2.4 .....

.....

2.5 .....

.....

2.6 .....

.....

.....

2.7 .....

2.8 .....

2.9 .....

2.10 .....

2.11 .....

.....

.....

.....

3

3.1 .....

3.2 .....

3.3 .....

3.4 .....

3.5 .....

.....

3.6 .....

.....

.....

3.7 .....

.....

.....

.....

3.8 .....

.....

3.9 .....

3.10 .....

.....

3.11 .....

3.12 .....

3.13 .....

3.14 .....

3.15 .....

3.16 .....

3.17 .....

3.18 .....

.....

3.19 .....

.....

.....

.....