

LAB-12 (SSH, HTTPS)

Ex. 1 | HTTP Authentication

1. 401 Authorization Required
2. Υπάρχει και το πεδίο 'Authorization'
3. Authorization: Basic ZWR1LWR5OnBhc3N3b3Jk
4. Source data : edu-dy:password
5. Ότι παρέχει μία πολύ απλού τύπου κρυπτογράφηση την οποία μπορεί να "σπάσει" ο οποιοσδήποτε.

Ex. 2 | SSH - Secure SHell

1. TCP
2. Ports : 63232 (mine), 22 (server)
3. Η θύρα 22
4. Display filter : ssh
5. Έκδοση πρωτοκόλλου : SSH-2.0, Έκδοση λογισμικού : OpenSSH_6.6.1_hpn13v11, Σχόλια : FreeBSD-20140420
6. Έκδοση πρωτοκόλλου : SSH-2.0, Έκδοση λογισμικού : PuTTY_Release_0.74. Όχι, δεν περιλαμβάνονται σχόλια
7. Υπάρχουν 11 αλγόριθμοι. Πρώτοι δύο : 1. curve25519-sha256@libssh.org, 2. ecdh-sha2-nistp256
8. Υπάρχουν 6 αλγόριθμοι. Πρώτοι δύο : 1. ssh-ed15519, 2. ecdsa-sha2-nistp256
9. 1. aes256-ctr, 2. aes256-cbc
10. 1. hmac-sha2-256, 2. hmac-sha1
11. 1. zlib, 2. zlib@openssh.com
12. Θα ακολουθήσουν τον : curve25519-sha256@libssh.org. Το wireshark τον εμφανίζει στο πεδίο 'Key Exchange'
13. Θα χρησιμοποιηθεί ο : aes256-ctr
14. Θα χρησιμοποιηθεί ο : hmac-sha2-256
15. Θα χρησιμοποιηθεί ο : zlib@openssh.com
16. Όχι, δεν φαίνεται να τους εμφανίζει
17. Τους 'Elliptic Curve ...', 'New Keys' και 'Encrypted packet'
18. Όχι, δεν μπορούμε να τα εντοπίσουμε γιατί αφού οι δύο πλευρές συμφώνησαν σε αλγόριθμο κρυπτογράφησης, όλα τα υπόλοιπα πακέτα είναι κρυπτογραφημένα
19. Συγκρίνοντάς το κυρίως με το telnet που έχουμε δει, μπορεί λόγω του public key να επιβεβαιώσει την αυθεντικότητα των δύο πλευρών και καθώς παρέχει

DIRECTORY

Ex. 1 | HTTP Authentication

Ex. 2 | SSH - Secure SHell

Ex. 3 | HTTPS

DICTIONARY

SSH (Secure SHell)

Πρωτόκολλο για ανταλλαγή δεδομένων μέσω ασφαλούς διαύλου

TLS (Transport Layer Security)

Πρωτόκολλο ασφαλούς μεταφοράς (πρίν το στρώμα εφαρμογής, μετά το στρώμα μεταφοράς)


SSL (Secure Sockets Layer)

Ήταν ο πρόγονος του TLS, δημιουργήθηκε σαν πρωτόκολλο το 1995 και έφτασε μέχρι το SSLv3

cipher suite

Σύνολο αλγορίθμων που ανταλλάσσουν οι δύο πλευρές στο https (αλγόριθμοι key, authentication, encryption, MAC)

NOTES


 Το SSH χρησιμοποιεί public key cryptography για τη πιστοποίηση

κρυπτογράφηση με πολλούς διαφορετικούς αλγορίθμους το καθιστά πολύ πιο ασφαλές για την διάρθρωση των δεδομένων.

Ex. 3 | HTTPS

1. Capture filter : host 147.102.40.19
2. Display filter : tcp.seq == 0 and tcp.ack == 0
3. Στις θύρες 80 και 443
4. 80 → http, 443 → https
5. 6 για http και 2 για https
6. 64493 και 64494
7. Content Type (1 byte), Version (2 bytes), Length (2 bytes)
8. Handshake (22), Change Cipher Spec (20), Application Data (23)
9. Client Hello (1), Server Hello (2), Certificate (11), Client Key Exchange (16), New Session Ticket (4)
10. 2 μηνύματα, όσα και οι συνδέσεις HTTPS
11. Version: TLS 1.0
12. 32 bytes. 4 πρώτα bytes : 00 a1 c5 ae. Παριστάνουν μία ημερομηνία (May 3, 1970)
13. Υποστηρίζει 16 cipher suites. 2 πρώτες : 0x8a8a, 0x1301
14. Θα χρησιμοποιηθεί TLS 1.2. Cipher Suite:
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f)
15. 32 bytes. 4 πρώτα random bytes : 50 0f 0a 05
16. Όχι, δεν φαίνεται να χρησιμοποιείται κάποια μέθοδος συμπίεσης
17. KEX : ECDHE, Authentication : RSA, Encryption : GCM, Hashing : SHA
18. 4278 bytes
19. 3 πιστοποιητικά. Έχουν μεγάλο όνομα ...
20. Χρειάστηκαν 4 πλαίσια
21. Τα μήκη των κλειδιών είναι 32 bytes. Pubkey client : 3a312..., Pubkey server : d7ab5
22. Έχει μήκος 6 bytes
23. 45 bytes
24. Ναι παρατηρήσαμε
25. Όχι, δεν παρατήρησα.
26. Πρόκειται για ένα TLS notification το οποίο στέλνεται για να ειδοποιήσει ότι το session μεταξύ των δύο συσκευών σταματάει καθώς δεν υπάρχουν άλλα δεδομένα να σταλούν.
27. Στη περίπτωση του HTTP μπορούμε εύκολα να βρούμε τη πληροφορία που ψάχνουμε, ενώ στο HTTPS δεν μπορούμε γιατί όλη η πληροφορία είναι

αυθεντικότητας του άλλου υπολογιστή

 Υπάρχει και η έκδοση 2 του SSH, η οποία περιέχει τα :

- SSH-TRANS
- SSH-AUTH
- SSH-CONN

κρυπτογραφημένη

28. Το HTTPS προσφέρει ακεραιότητα της πληροφορίας που ανταλλάσσεται, σε αντίθεση με το HTTP στο οποίο ο οποιοσδήποτε “ακροατής” μπορεί να υποκλέψει τα δεδομένα.