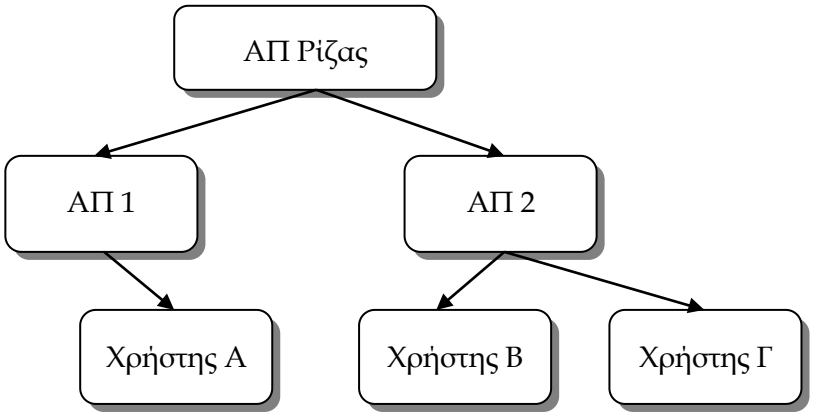
ΠΡΟΣΤΑΣΙΑ ΚΑΙ ΑΣΦΑΛΕΙΑ ΣΥΣΤΗΜΑΤΩΝ ΥΠΟΛΟΓΙΣΤΩΝ (Εργαστήριο)

Ενότητα 7

Ψηφιακά πιστοποιητικά και OpenSSL

Τα ψηφιακά πιστοποιητικά (digital certificates) αποτελούν το κύριο μέσο με το οποίο μια οντότητα μπορεί να πάρει με αξιόπιστο τρόπο ένα αυθεντικοποιημένο αντίγραφο του δημοσίου κλειδιού μιας άλλης οντότητας. Χρησιμοποιούνται σε όλες τις εφαρμογές όπου υπάρχουν ψηφιακές υπογραφές ή/και κρυπτογράφηση η οποία κάνει χρήση κρυπτογραφίας δημοσίου κλειδιού. Ενδεικτικές εφαρμογές είναι η προστασία emails (S/MIME) και η προστασία πρόσβασης σε ιστοσελίδες με τη χρήστη του SSL/TLS πρωτοκόλλου.

Ένα ψηφιακό πιστοποιητικό είναι μια ακολουθία δεδομένων η οποία υπογράφεται ψηφιακά από μια τρίτη έμπιστη οντότητα η οποία ονομάζεται Αρχή Πιστοποίησης (Certification Authority – CA). Η Αρχή Πιστοποίησης (ΑΠ) είναι υπεύθυνη για την έκδοση, ανανέωση και ανάκληση του πιστοποιητικού και γενικά για ότι έχει σχέση με τη διαχείριση του. Οι Αρχές Πιστοποίησης τυπικά εντάσσονται σε μια Υποδομή Δημοσίου Κλειδιού (Public Key Infrastructure – PKI) και συνήθως σε μια ιεραρχημένη αρχιτεκτονική η οποία έχει την παρακάτω μορφή.



Ένα ψηφιακό πιστοποιητικό περιλαμβάνει πληροφορίες για τον εκδότη καθώς και για τον κάτοχο του πιστοποιητικού όπως είναι το ονοματεπώνυμο του (αν πρόκειται για απλό χρήστη), το email του, τα στοιχεία του οργανισμού στον οποίο δραστηριοποιείται ο χρήστης (π.χ. IHU) κ.α. Το πρότυπο που έχει υιοθετηθεί σε παγκόσμιο επίπεδο για τη μορφή και τα περιεχόμενα των πιστοποιητικών είναι το X.509 το οποίο ορίζεται και στο RFC5280. Η πιο εύκολη πρόσβαση στα πιστοποιητικά που είναι αποθηκευμένα στον υπολογιστή του χρήστη είναι μέσω του browser.

Προκειμένου μια εφαρμογή να μπορέσει να επαληθεύσει το ψηφιακό πιστοποιητικό ενός χρήστη θα πρέπει να έχει στη διάθεση της το δημόσιο κλειδί της ΑΠ που έχει υπογράψει το πιστοποιητικό.

Δημιουργία αυτοϋπογραφόμενου πιστοποιητικού

Η δημιουργία μιας υποδομής δημοσίου κλειδιού μέσα στην οποία δραστηριοποιείται μια ΑΠ και από την οποία ο ενδιαφερόμενος χρήστης μπορεί να πάρει ψηφιακό πιστοποιητικό ξεκινά με τη δημιουργία των πιστοποιητικών των ΑΠ. Μια ΑΠ προκειμένου να μπορέσει να εκδίδει πιστοποιητικά για τρίτους θα πρέπει πρώτα να δημιουργήσει το δικό της ζεύγος κλειδιών και το αντίστοιχο πιστοποιητικό για το δημόσιο κλειδί της. Όταν πρόκειται για μια νέα Υποδομή Δημοσίου Κλειδιού θα πρέπει αρχικά να δημιουργήσουμε τη βάση της υποδομής, μια ΑΠ Ρίζας το πιστοποιητικό της οποίας

πρέπει να είναι αυτοϋπογραφόμενο, δηλαδή η ίδια η ΑΠ Ρίζας θα πρέπει να υπογράψει το δικό της πιστοποιητικό με το δικό της ιδιωτικό κλειδί καθώς δεν υπάρχει άλλη ΑΠ που να μπορεί να υπογράψει αυτό το πιστοποιητικό. Σε κάθε άλλη περίπτωση το πιστοποιητικό υπογράφεται από άλλη ΑΠ που βρίσκεται ψηλότερα στην ιεραρχία.

Επομένως, η δημιουργία της ιεραρχίας θα ξεκινήσει με τη δημιουργία αυτοϋπογραφόμενου πιστοποιητικού για την ΑΠ Ρίζας η οποία γίνεται με τη χρήση της εντολής `req`. Ο χρήστης καλείται να απαντήσει σε αρκετές ερωτήσεις που αφορούν την ΑΠ Ρίζας και οι οποίες θα συμπεριληφθούν στο πιστοποιητικό, όπως:

Country, State, Organization, Common Name και Email Address.

Δε χρειάζεται να δοθούν τιμές σε όλα τα πεδία. **Ωστόσο τα πεδία Country Name** (δηλώνει τη χώρα), **State** (δηλώνει περιοχή στην χώρα), **Organization Name** (δηλώνει τον οργανισμό με τον οποίο σχετίζεται ο χρήστης), και **Common Name** (δηλώνει το όνομα με το οποίο είναι γνωστός ο κάτοχος του πιστοποιητικού)**, δε θα πρέπει να είναι κενά**.

Τα παραπάνω στοιχεία μπορούν να αλλάξουν τροποποιώντας τις αντίστοιχες παραμέτρους στο αρχείο παραμετροποίησης openssl.cfg το οποίο βρίσκεται στα αρχεία της εγκατεστημένης εφαρμογής.

Σημείωση: για τη διευκόλυνση σας στην εκτέλεση των παρακάτω εντολών συνιστάται να δημιουργήσετε στην επιφάνεια εργασίας φάκελο με το όνομα «openssl» στον οποίο θα αντιγράψετε το αρχείο openssl.cfg. Επιπλέον, η εκτέλεση των εντολών θα γίνεται από command prompt αφού πρώτα μεταβείτε στον συγκεκριμένο φάκελο, και ακολουθώντας τη μέθοδο της εκτέλεσης των εντολών εκτός περιβάλλοντος openssl, με τη χρήση ωστόσο του προθέματος openssl πριν από κάθε εντολή. Για παράδειγμα δείτε την επόμενη εντολή η οποία είναι και η πρώτη του σεναρίου μας.

```
# Δημιουργία αυτοϋπογραφόμενου κλειδιού μαζί με νέο ζεύγος κλειδιών.
# Το κλειδί είναι τύπου rsa και έχει μέγεθος 2048 bits.
C:\Users\user\Desktop\openssl>openssl req -x509 -nodes -days 3650 -
newkey rsa:2048 -keyout CAkey.pem -out CAcert.pem –config openssl.cfg
```

Με την ολοκλήρωση της εκτέλεσης της εντολής δημιουργείται ζεύγος κλειδιών το οποίο αποθηκεύεται στο αρχείο `CAkey.pem` και αυτοϋπογραφόμενο πιστοποιητικό το οποίο αποθηκεύεται στο αρχείο `CAcert.pem`. Η παράμετρος `–x509` χρησιμοποιείται για να δηλώσει ότι θα πρέπει ταυτόχρονα να δημιουργηθεί ένα αυτοϋπογραφόμενο πιστοποιητικό (self-signed certificate). Η παράμετρος `–days` για ποιο λόγο χρησιμοποιείται;

Σημείωση: Το κύριο χαρακτηριστικό του πιστοποιητικού της Αρχής Πιστοποίησης Ρίζας είναι ότι τα στοιχεία του εκδότη (Issuer) είναι ακριβώς ίδια με τα στοιχεία του κατόχου (Subject).

Μπορείτε να δείτε το αυτοϋπογραφόμενο πιστοποιητικό αλλάζοντας την κατάληξη του από .pem σε .cer.

Δημιουργία αίτησης από το χρήστη

Κάθε άλλη οντότητα πλην της ΑΠ Ρίζας προκειμένου να πάρει ένα πιστοποιητικό θα πρέπει να δημιουργήσει μια αίτηση για υπογραφή δημοσίου κλειδιού την οποία θα στείλει στην ΑΠ της επιλογής της για την έκδοση του ψηφιακού πιστοποιητικού.

Σημαντικό: **Παρόλο που δε συνηθίζεται η ΑΠ Ρίζας να υπογράφει πιστοποιητικά για τελικούς χρήστες για τις ανάγκες του εργαστηρίου ο χρήστης θα ζητήσει από την ΑΠ Ρίζας να του εκδώσει ένα πιστοποιητικό.**

Η εντολή που θα χρησιμοποιηθεί είναι η ίδια με αυτή που χρησιμοποίησε η ΑΠ Ρίζας για να δημιουργήσει το αυτοϋπογραφόμενο πιστοποιητικό πλην της παραμέτρου `–x509`. Ο χρήστης καλείται

πάλι να απαντήσει σε αρκετές ερωτήσεις που αφορούν τον κάτοχο του πιστοποιητικού και οι οποίες θα συμπεριληφθούν στο πιστοποιητικό, όπως:

Country, State, Organization, Common Name και Email Address.

Για το Common Name θα βάλετε το όνομα σας ενώ για το Email Address τη διεύθυνση email σας. Δε χρειάζεται να δοθούν τιμές σε όλα τα πεδία. **Ωστόσο τα πεδία Country Name** (δηλώνει τη χώρα) **Common Name** (δηλώνει το όνομα με το οποίο είναι γνωστός ο κάτοχος του πιστοποιητικού) **και Organization Name** (δηλώνει τον οργανισμό με τον οποίο σχετίζεται ο χρήστης) **δε θα πρέπει να είναι κενά**. **Το Organization Name στο πιστοποιητικό του χρήστη πρέπει να είναι το ίδιο με το Organization Name στο πιστοποιητικό της ΑΠ.**

```
# Δημιουργία νέας αίτησης μαζί με νέο ζεύγος κλειδιών. Το κλειδί  #
είναι τύπου rsa και έχει μέγεθος 1024 bits.
C:\Users\user\Desktop\openssl>openssl req –newkey rsa:1024 –keyout
mykey.pem –out myreq.pem –config openssl.cfg
```

Το ιδιωτικό κλειδί αποθηκεύεται στο αρχείο `mykey.pem` ενώ η αίτηση στο αρχείο `myreq.pem`. **Το κλειδί προστατεύεται με κωδικό που θα πρέπει να ορίσει ο χρήστης.**

Μέσω της ίδιας εντολής (req) ο χρήστης μπορεί να ελέγξει τα περιεχόμενα της αίτησης καθώς και την υπογραφή πάνω σε αυτή

```
# Έλεγχος της υπογραφής πάνω στην αίτηση υποδεικνύοντας το κλειδί που
έχει χρησιμοποιηθεί για την υπογραφή
C:\Users\user\Desktop\openssl>openssl req –in myreq.pem –noout -text –
verify –key mykey.pem –config openssl.cfg

# Έλεγχος των περιεχομένων της αίτησης
C:\Users\user\Desktop\openssl>openssl req –in myreq.pem –noout –text –
config openssl.cfg
```

Κατόπιν μπορούμε να δώσουμε την αίτηση στην Αρχή Πιστοποίησης για να την υπογράψει και να μας δώσει το πιστοποιητικό. Η αίτηση μπορεί να αποσταλεί με οποιονδήποτε τρόπο, π.χ. email.

Υπογραφή πιστοποιητικού

Για τις ανάγκες της έκδοσης του πιστοποιητικού, η ΑΠ θα πρέπει να έχει ορίσει κάποιες παραμέτρους που σχετίζονται με το χρόνο ζωής του πιστοποιητικού, το σειριακό αριθμό, ποιο κλειδί θα χρησιμοποιηθεί για την υπογραφή κ.α. τις οποίες τυπικά τις αποθηκεύει σε ένα configuration αρχείο.

Η εφαρμογή έχει εξ ορισμού ένα αρχείο παραμέτρων (`openssl.cfg`) το οποίο χρησιμοποιεί για την περίπτωση που ο χρήστης δε δώσει το δικό του αρχείο ή δε δηλώσει άλλες παραμέτρους. Στο πλαίσιο του εργαστηρίου θα χρησιμοποιήσουμε αυτό το εξ ορισμού αρχείο του Openssl αλλά αντί να τροποποιήσουμε τις τιμές στο αρχείο θα περάσουμε τις νέες τιμές από τη γραμμή εντολών. Επιπλέον θα πρέπει να δημιουργήσουμε δύο ακόμη αρχεία:

- Το αρχείο `serial` (για τις ανάγκες του εργαστηρίου δημιουργείται στο `.\demoCA`) στο οποίο αποθηκεύεται ο σειριακός αριθμός του πιστοποιητικού και ο οποίος θα αλλάζει με την υπογραφή ενός πιστοποιητικού. **Θα πρέπει αρχικά να περιλαμβάνει μια τιμή της επιλογής σας**. (Ο αριθμός που θα βάλετε στο αρχείο αυτό είναι σε δεκαεξαδική μορφή οπότε και θα πρέπει να αποτελείται από ζυγό αριθμό ψηφίων, π.χ. 01). ΠΡΟΣΟΧΗ: μετά την πληκτρολόγηση του αριθμού αλλάξτε γραμμή, σώστε το αρχείο και κλείστε το.

- Το αρχείο `index.txt` (για τις ανάγκες του εργαστηρίου δημιουργείται στο `.\demoCA`) στο οποίο η ΑΠ κρατάει πληροφορίες για τα πιστοποιητικά που έχει δημιουργήσει. **Θα πρέπει αρχικά να είναι κενό (θα πρέπει ωστόσο να το δημιουργήσετε)**. ΠΡΟΣΟΧΗ: Μην αφήσετε

κάποια κενή γραμμή στο αρχείο γιατί θα παρουσιαστεί λάθος κατά την εκτέλεση της εντολής openssl ca.

Η υπογραφή της αίτησης γίνεται με τη χρήση της εντολής `ca`.

```
# Δημιουργία και υπογραφή πιστοποιητικού βάσει των στοιχείων που #
υπάρχουν στην αίτηση myreq.pem του χρήστη και αποθήκευση του
# πιστοποιητικού με το όνομα mycert.pem στο ίδιο directory.
# Για την υπογραφή θα πρέπει να δηλώσουμε και το κλειδί της ΑΠ που
# θα χρησιμοποιηθεί για αυτό.
C:\Users\user\Desktop\openssl>openssl ca –keyfile CAkey.pem –
cert CAcert.pem –outdir ./ -in myreq.pem –out mycert.pem –
config openssl.cfg
```

Κατά τη διάρκεια εκτέλεσης της εντολής παρατηρούμε αν τα δεδομένα που πρόκειται να υπογραφούν είναι τα ίδια με αυτά που ζητήσαμε ως χρήστες.

Αφού ολοκληρωθεί η έκδοση του πιστοποιητικού μπορούμε να δούμε τις τιμές του πιστοποιητικού σε μορφή text με την εντολή `x509`.

```
# Προβολή διαφόρων τιμών του πιστοποιητικού.
C:\Users\user\Desktop\openssl>openssl x509 -text -in mycert.pem
```

Το PKCS#12 αποτελεί ένα διεθνώς αναγνωρισμένο και αποδεκτό από όλες τις εφαρμογές πρότυπο για τη διακίνηση και διαχείριση ψηφιακών πιστοποιητικών. Προκειμένου να μπορούμε να διαχειριστούμε πιο εύκολα τα ψηφιακό μας πιστοποιητικό και να μπορέσουμε να το εισάγουμε (import) από όλες τις εφαρμογές θα πρέπει να το αποθηκεύσουμε σε μορφή PKCS#12 με τη χρήση της εντολής `pkcs12`. Για τη αποθήκευση ενός πιστοποιητικού σε μορφή PKCS#12 ο χρήστης χρειάζεται να έχει το πιστοποιητικό και το ιδιωτικό κλειδί.

```
# Αποθήκευση πιστοποιητικού σε μορφή PKCS#12
C:\Users\user\Desktop\openssl>openssl pkcs12 –export –out mycert.pfx –
inkey mykey.pem –in mycert.pem –name "My Certificate"
```

Κατά τη διάρκεια εκτέλεσης της εντολής θα σας ζητηθεί τόσο ο κωδικός που είχατε βάλει αρχικά για να προστατεύσετε το ιδιωτικό σας κλειδί όσο και ένας νέος κωδικός που θα χρησιμοποιηθεί για την προστασία του PKCS#12 αρχείου.

Άσκηση 1

Μετονομάστε τα πιστοποιητικά `CAcert.pem` σε `CAcert.cer` και `mycert.pem` σε `mycert.cer` και ανοίξτε τα. Για ποιο λόγο το πιστοποιητικό εμφανίζεται ως μη έγκυρο;

Εγκαταστήστε το πιστοποιητικό της ΑΠ Ρίζας στον υπολογιστή σας και ελέγξτε εκ νέου το πιστοποιητικό σας.

Άσκηση 2

Στο πλαίσιο της άσκησης θα εισάγουμε στον browser firefox τόσο το πιστοποιητικό της ΑΠ Ρίζας όσο και το δικό μας πιστοποιητικό ώστε να δούμε τα δύο πιστοποιητικά με τον τρόπο με τον οποίο τα παρουσιάζει ο firefox.

Τα πιστοποιητικά στον firefox είναι προσβάσιμα μέσα από το μενού Options → Advanced → Certificates → View Certificates. Από εκεί μπορούμε να εισάγουμε κάποιο πιστοποιητικό και γενικά να διαχειριστούμε τα πιστοποιητικά τόσο των ΑΠ όσο και τα δικά μας.

Αρχικά εισάγετε το δικό σας πιστοποιητικό σας και ελέγχετε εάν μπορεί ο browser να επαληθεύσει την εγκυρότητα του.

Άσκηση 3

Χρησιμοποιώντας τα πιστοποιητικά που δημιουργήσατε κάνετε όλες τις απαραίτητες ενέργειες ώστε να στείλετε το ακόλουθο μήνυμα προς παραλήπτη της επιλογής σας:

$$E_k(m), Sign(m), E_{Pub\_B}(k)$$

Όπου,

m: μήνυμα προς αποστολή

$E_k(x)$: κρυπτογράφηση του *x*, με τον αλγόριθμο *E* και κλειδί το *k*.

Sign(x): υπογραφή στο μήνυμα *x*.

$E_{Pub\_B}(x)$: κρυπτογράφηση του *x* με τη χρήση του δημοσίου κλειδιού *Pub_B* της οντότητας *B*.

ΠΡΟΣΟΧΗ: Για τις ανάγκες της άσκησης θα δημιουργήσετε ζεύγη κλειδιών και πιστοποιητικά για 2 χρήστες. Οι χρήστες θα πάρουν πιστοποιητικό απευθείας από την Αρχή Πιστοποίησης Ρίζας (δε χρειάζεται να δημιουργηθεί ενδιάμεση αρχή πιστοποίησης).