به نام خدا

امضاي ديجيتال

پروژه پايانترم درس تئوري رمزنگاري
دکتر سلماسي زاده

ابوالحسن شمسائي
دانشجوي کارشناسي ارشد فناوري اطلا عات
shamsaie@mehr.sharif.edu

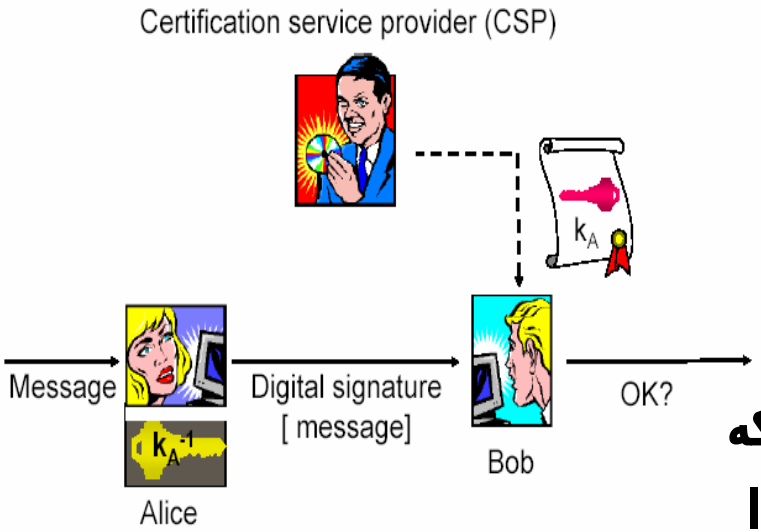
رئوس مطالب

- مقدمه
  - تئوري
  - قوانين در امضاي ديجيتال
  - امضاي مبتي به خدمتگزار
- تعدادي از استانداردهاي معتبر امضاي ديجيتال
- استاندارد امضاي ديجيتال (DSS)
  - الگوريتم امضاي ELGamal
  - الگوريتم امضاي Schnorr
  - الگوريتم امضاي ديجيتال (DSA)
- پياده سازي نرم افزاري و سخت افزاري DSS
- كاربردها
- نتايج
- منابع

مقدمه

- در بسياري از برنامه هاي كاربردي وجود يك سرويس انكار ناپذيرلازم ديده مي شود .
- امضاي ديجيتال مي تواند مكانيزم امنيتي لازم را فراهم كند .
- كاربرد در مواردي چون تأييد صلاحيت و يا جامعيت سرويس ها .
- امضاي ديجيتال به عنوان پيش نياز اساسي تجارت الكترونيك .
- الگوريتم هاي بسياري در اين راستا مطرح شده اند .

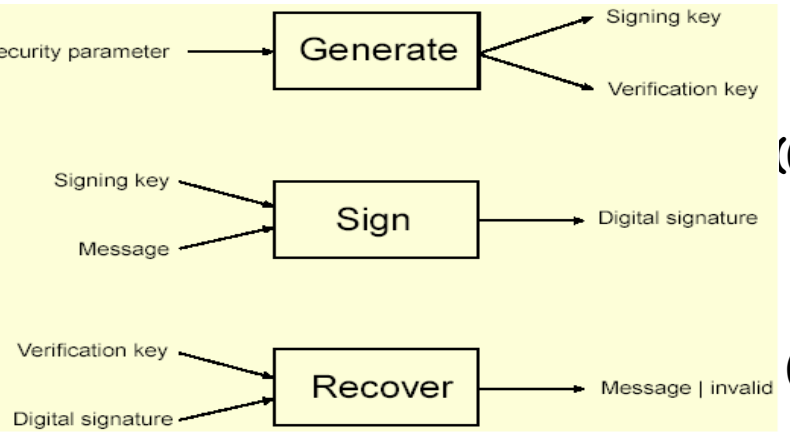
تئوري



- يك امضاي ديجيتال
  - با توجه به ISO/IEC 7498-2
  - داده افزوده شده و يا يك تبديل نهفته
  - از يك واحد داده
  - به دريافت كننده اين امكان را مي دهد كه صحت و سقم و جامعيت آن واحد داده را بررسي و كنترل كند.
- انواع مختلف امضاي ديجيتال
  - امضاي ديجيتال بصورت داده افزوده
  - امضاي ديجيتال بصورت بازيافت پيام

امضاي ديجيتال بصورت بازيافت پيام

- ۳ الگوريتم :



– Generate (1^n)

- توليد جفت كليد($(k, k^{-1}) = Generate\ (1^n)$)

– Sign (k^{-1}, m)

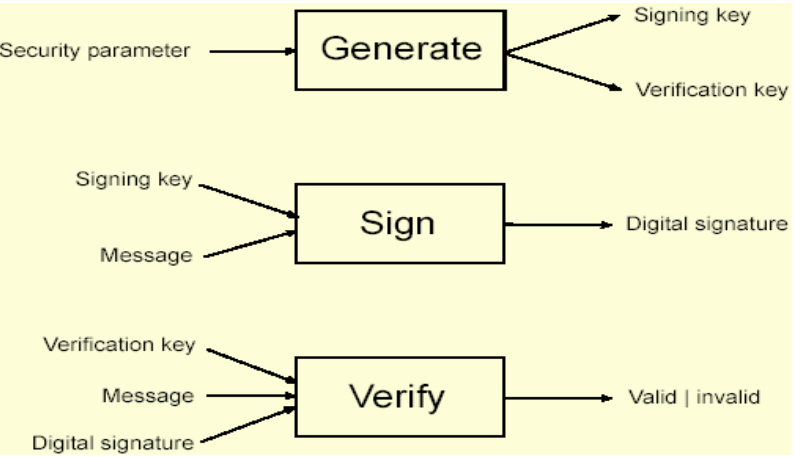
- امضاكردن يك پيام ($s = Sign(k^{-1}, m)$)

– Recover (k, s)

- بازيابي پيام ($m = Recover\ (k, s)$)

امضاء دیجیتال با داده افزوده

- ۳ الگوریتم :



– Generate (1^n)

- تولید جفت کلید($(k, k^{-1}) = Generate\ (1^n)$)

– Sign (k^{-1}, m)

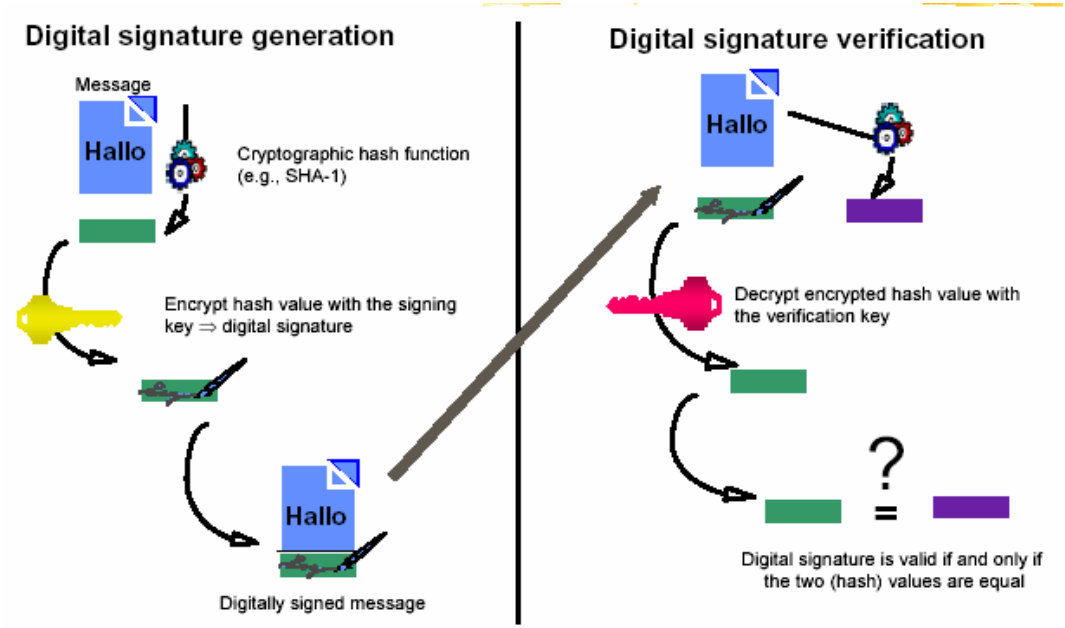
- امضاکردن یك پیام ($s = Sign(k^{-1}, m)$)

– Verify (k ,m ,s)

- تعیین صحت امضا ($output= Verify\ (k ,m ,s)$)

تابع در هم ساز h

- در بیشتر موارد تابع Sign از تابع در هم سازh استفاده مي كند $(s = sign(k^{-1}, h(m)))$

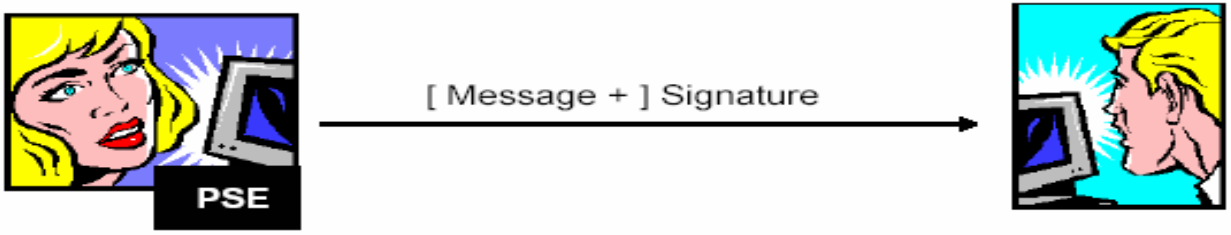


قوانين در امضاء ديجيتال

- در اواخر دهه ۱۹۹۰ امضا ديجيتال براي تجارت الكترونيك به عنوان يك پيش نياز قانوني مورد بحث و بررسي قرار گرفت.
- كشورهاي بسياري با فعاليت در اين زمينه استاندارد، اسناد و قوانين خاصي را براي امضاي ديجيتالي وضع كردند.
  - SigG در آلمان (۱۹۹۷)
  - E-SIGN در آمريكا
  - ZertEs در سوئد (۲۰۰۵)
- قوانين و استانداردهاي ملي بايد بتواند با هم تعامل كند.
  - مصوبه ۱۹۹۹/۹۳/EC در سال ۱۹۹۹ توسط پارلمان اروپا.
- امروزه هر كشوري قوانين مربوط به امضاي ديجيتال خود را دارد.
- قوانين هيچ كشوري كاملا موفق نبوده است.

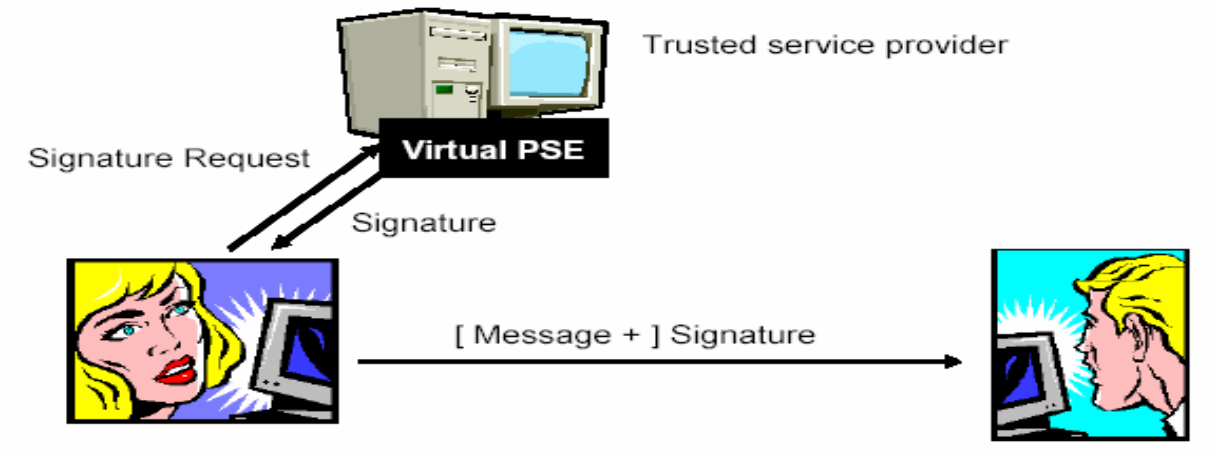
Client side امضاي

- مشكلات امضاي Client side
  - نيازمندي به سرويس هاي تصديق كننده(PKI Deployment)
  - نيازمندي به سادگي در جابجايي
  - احتمال بي ارزش بودن امضاء بر خلاف فرض اوليه



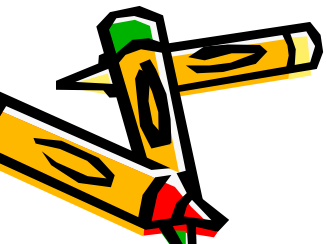
Server side امضا ي

- مزاياي امضا ي Server side
  - انعطاف پذيري
    - انواع امضاهاي ديجيتال
    - فرمتهاي مختلف امضاي ديجيتال مثل امضاي مبتني بر XML
  - تطبيق آسان با نيازهاي جديدمثل اضافه شدن يك تابع در هم ساز جديد
  - امكان تهيه سرويس هاي مكمل مثل سرويس برچسب زمان
- معايب امضا ي Server side
  - وابستگي به وجود شبكه
  - كنترل دور كليد امضا



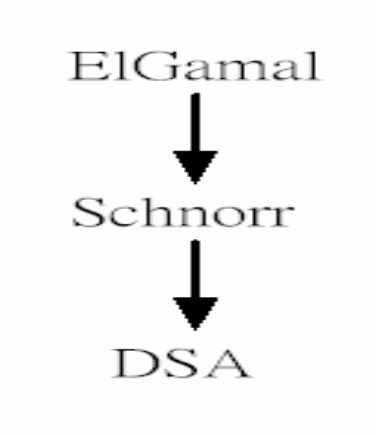
استانداردهاي معتبر امضاي ديجيتال

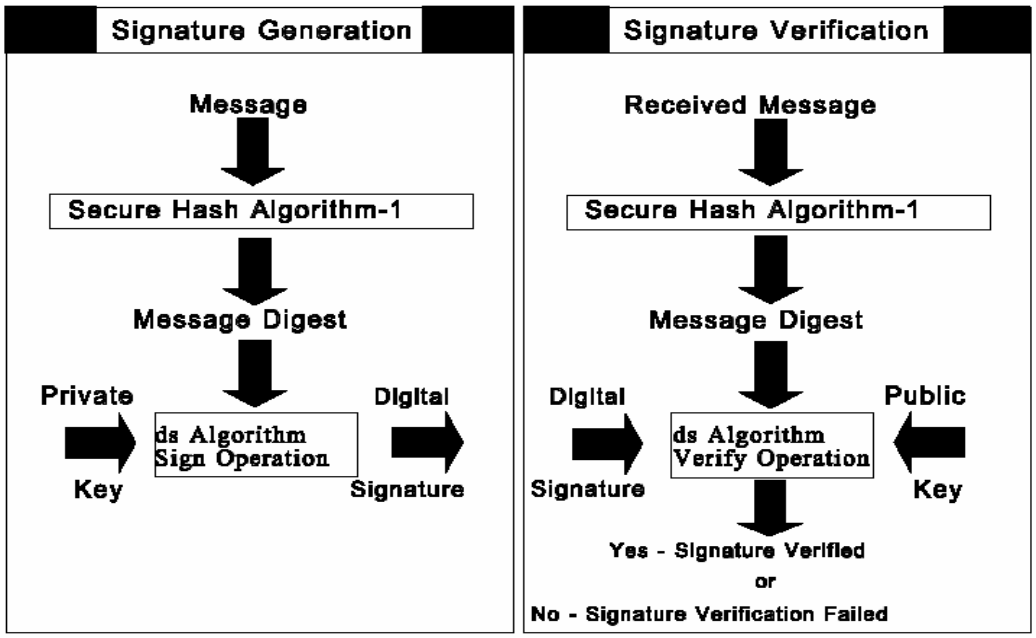
- استاندارد ملي آمريكا:DSS
- استاندارد ملي روسيه:GOST
- استاندارد ملي ژاپن :ESIGN
- استاندارد غيررسمي:RSA
- استاندارد RSA:ISO/ICE9796
- استاندارد X9.30-199 ملي آمريكا: ElGamal و RSA



استاندارد امضاي ديجيتال (DSS)

- انتشار DSA توسط موسسه ملي استاندارد و تكنولوژي آمريكا(NIST) در سال ۱۹۹۱
- استاندارد امضاي ديجيتال (DSS) بر اساس آن بنا نهاده شده است
- شباهت اين الگوريتم با الگوريتم ELGamal





الگوريتم امضاي ElGamal(۱)

- کمي رياضيات:
  - قضيه: دو عدد n, a بگونه اي که GCD(a ,n)=1 را فرض کنيد آنگاه رابطه زير وجود دارد

$$a^{x} \pmod{n} \equiv a^{x \pmod{\varphi(n)}} \pmod{n}$$

  - اثبات قضيه :

Let $y = x \pmod{\varphi(n)} \Rightarrow x = k \times \varphi(n) + y$ for some k.

$$a^{x} \pmod{n} = a^{k \times \varphi(n) + y} \pmod{n}$$

$$\because a^{\varphi(n)} \pmod{n} = 1 \Rightarrow a^{k \times \varphi(n)} \pmod{n} = 1$$

$$\therefore a^{x} \pmod{n} = a^{x} \times 1 \pmod{n} = a^{x} \times a^{k \times \varphi(n)} \pmod{n}$$

$$= a^{y} \pmod{n} = a^{x \pmod{\varphi(n)}} \pmod{n}$$

الگوريتم امضاي ElGamal(۲)

- توليد كليد:
  - يك عدد بزرگ اول p و يك ريشه اوليه g در Z_p^* را انتخاب كنيد و آنها را منتشر كنيد.
  - يك عدد x را در Z_{p-1} انتخاب كنيد.
  - مقدار y را از رابطه $y \equiv g^x \pmod p$ بدست آوريد.
  - كليد عمومي y و كليد خصوصي x مي باشد

الگوريتم امضاي ElGamal(۳)

- انجام امضا:
  - امضا کننده به طور تصادفي عدد K را در Z_{p-1} انتخاب مي کند.
  - مقدار r را از رابطه $r = g^k \pmod p$ محاسبه مي کند.
  - مقدار s را از رابطه $s=k^{-1} \times (M- xr) \pmod{p-1}$ محاسبه مي کند.
  - آنگاه زوج مقدار(s ، r) =S امضاي ديجيتال پيام M خواهند بود.

الگوريتم امضاي ELGamal(۴)

- تاييد امضا:
  - اگر $g^M = y^{r} \times r^{s} \pmod p$ در آنصورت
    - امضاء درست مي باشد.
    - در غير اينصورت امضاء نادرست است.

الگوريتم امضاي Schnorr (۱)

- توليد كليد:
  - يك عدد اول p بزرگ $p \ge 2^{512}$ را انتخاب كنيد.
  - يك عدد اول q به طوري كه $q|p-1$ و $q \ge 2^{160}$ را انتخاب كنيد.
  - ريشه g را در Z_p^* بگونه اي كه $g \ne 1$ باشد و $g = h^{(p-1)/q}$ را انتخاب كنيد.
  - p ، q و g را انتشار دهيد.
  - عدد x را در Z_q انتخاب كنيد.
  - مقدار y را از رابطه $y \equiv g^x \pmod{p}$ محاسبه كنيد.
  - y كليد عمومي و x كليد خصوصي خواهد بود.

الگوريتم امضاي Schnorr(۲)

- انجام امضا:
  - امضا كننده به طور تصادفي عدد K را در Z_q انتخاب مي كند.
  - مقدار r را از رابطه $r=g^k \pmod p$ محاسبه مي شود.
  - مقدار e را از رابطه $e = h(r, m)$ محاسبه مي شود كه h يك تابع درهمساز يكطرفه مي باشد.
  - مقدار s را از رابطه $s=k - xe \pmod q$ محاسبه مي كند.
  - آنگاه زوج مقدار $S=(e, s)$ امضاي ديجيتال آن پيام خواهند بود.

الگوريتم امضاي Schnorr(۳)

- تاييد امضا:
  - مقدار r را از رابطه $r = g^s \times y^e \pmod p$ محاسبه مي شود.
  - اگر $e = h(r, m)$ باشد در آن صورت امضا درست مي باشد.
  - در غيراينصورت امضا نادرست مي باشد.

الگوريتم امضای ديجيتال (DSA)(1)

- توليد كليد:
  - يك عدد اول p (۵۱۲ تا ۱۰۲۴ بيتي) انتخاب كنيد.
  - يك عدد اول q (۱۶۰بيتي) كه q|p-1انتخاب كنيد.
  - ريشه g را در Z_p^*بگونه اي كه $g \neq 1$ باشد و $g = h^{(p-1)/q}$ انتخاب كنيد.
  - q،p وg را انتشار دهيد.
  - عدد xرا در Z_q انتخاب كنيد.
  - عدد y را از رابط $y \equiv g^x \pmod p$محاسبه كنيد.
  - كليد عمومي y و كليد خصوصي x خواهد بود.

الگوريتم امضای ديجيتال (DSA)(2)

- انجام امضا:
  - امضا کننده عدد k را در z_q بطور تصادفي انتخاب مي کند.
  - عدد r را از رابطه $r = g^k \pmod p \pmod q$ محاسبه مي کند.
  - عدد s را از رابطه $s=k^{-1}\times(SHA\text{-}1(M) + xr) \pmod q$ محاسبه مي کند.
  - امضاي S جفت مقدار $S=(r , s)$ خواهند بود.

الگوريتم امضای ديجيتال (DSA)(3)

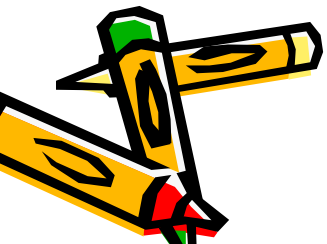
- تاييد امضا:
  - اگر $0<r<q$ و $0<s<q$ باشد به مرحله بعد مي رويم در غيراينصورت امضا نادرست مي باشد.
  - مقدار t را از رابطه $t = SHA\text{-}1(M) \times s^{-1} \pmod q$ محاسبه مي كنيم .
  - مقدار u را از رابطه $u = r \times s^{-1} \pmod q$ بدست آوريم .
  - اگر رابطه $r=g^{t}xy^{u} \pmod q \pmod p$ درست باشد درآن صورت امضا درست مي باشد.
  - در غير اين صورت امضا نادرست است.

نكات مثبت و نكات منفي DSS

- نكات مثبت DSS :
  - طول امضاي كوتاهتر مي باشد.
  - توليد كليد سريعتر صورت مي گيرد.
  - در صورتي كه امضا كننده تعداد زيادي r را محاسبه و ذخيره كند هزينه زمان امضا بسيار كم مي شود.
  - اين امضا توسط دولت آمريكا تاييد شده است .
- نكات منفي DSS:
  - DSS با RSA سازگار نمي باشد.
  - اگر s برابر صفر باشد نتيجه بررسي صحت امضاء نادرست خواهد بود.
  - فرآيند تعيين صحت امضا ۱۰۰ مرتبه كندتر از RSA مي باشد.

پياده سازي نرم افزاري و سخت افزاري DSS

- DSS پياده سازي هاي سخت افزاري چنداني ندارد.
- به عبارتي پياده سازي هاي موجود اغلب نرم افزاري هستند و در قسمت كوچكي از الگوريتم DSA از سخت افزار استفاده مي كنند .
- نمونه هايي از پياده سازي سخت افزاري و نرم افزاري را در ادامه خواهيم ديد.



پياده سازي نرم افزاري(۱)

- Java Cryptography Extension (JCE)يك مجموعه از package ها مي باشد كه بستر لازم را براي پياده سازي برنامه هاي كاربري كه نياز به موارد زير دارند فراهم مي كند.
  - رمزنگاري
  - توليد كليد
  - توافق بر كليد
  - الگوريتم كد تعيين اصالت پيام (MAC)
  - رمزنگاري هاي مختلف از قبيل رمز نگاري متقارن ، نامتقارن، قطعه اي و دنباله اي را پشتيباني مي كند.
- نمونه خاصي از پياده سازي نرم افزاري DSA را د رادامه خواهيم ديد كه البته فقط تابع Signآن انتخاب شده است .

پیاده سازي نرم افزاري(۲)

```
public void signMethod()
        {
                try {
                        KeyPairGenerator keyGen = KeyPairGenerator.getInstance("DSA", "SUN");
                        SecureRandom random = SecureRandom.getInstance("SHA1PRNG", "SUN");
                        keyGen.initialize(1024, random);
                        KeyPair pair = keyGen.generateKeyPair();
                        PrivateKey priv = pair.getPrivate();
                        PublicKey pub = pair.getPublic();
//Generating P , Q and G
                        P = new BigInteger(p1,16);
                        Q = new BigInteger(q1,16);
                        G = new BigInteger(g1,16);
// Generating x , y and K
                do {
                        x = generate_rand();
                        X = x.abs();
                        k = generate_rand();
                        K = k.abs();
                        xcmp = X.compareTo(Q);
                        kcmp = K.compareTo(Q);
                } while (xcmp != -1 && kcmp != -1);
                Y = G.modPow(X,P);
// Signing the message m
                R = (G.modPow(K,P)).mod(Q);
                Rtxt.setText(R.toString());
                BigInteger Kinv = K.modInverse(Q);
                BigInteger Sint1 = X.multiply(R);
                BigInteger Sint2 = hshm.add(Sint1);
                BigInteger Sint3 = Kinv.multiply(Sint2);
                S = Sint3.mod(Q);
                textField2.setText(S.toString());
                textField1.setText(Y.toString());
                }
                catch (Exception e) {
                        System.err.println("Caught exception " + e.toString());
                }
                }
// signMethod()
```

پیاده سازي سخت افزاري

- Platform ي بنام OS/390(z/OS)
  - IBM با گسترش JCE محصول جديدي بنام IBMJCE4758 را ساخته است.
  - اين محصول امكان استفاده از سخت افزار مشابه آنچه در JCE براي استفاده از نرم افزار است را می دهد.
  - ارتباط با OS/390(z/OS) كه سخت افزارپياده كننده بعضي از متد ها مي ياشد نيازمند CCA كه يك واسط براي ارتباط با سخت افزار مي باشد هست.
- پياده سازي سخت افزاري كاملي از DSA وجود ندارد.

کاربرد ها

- هر برنامه کاربردي که نيازمند جامعيت پيام و اصالت فرسنده باشد مي تواند از DSS استفاده کند.
- در سيستم هاي نرم افزاري توزيع شده کاربرد دارد.
- در ذخيره اطلاعات مي تواند مفيد باشد.
- در جامعيت پايگاه هاي داده مورد استفاده قرار مي گيرد.
- در برنامه هاي کاربري چون EMAIL مورد استفاده قرار مي گيرد.
- در نقل و انتقالات الکترونيکي پول مورد استفاده قرار مي گيرد.
- در نقل و انتقالات الکترونيکي داده مورد استفاده قرار مي گيرد.
- کاربرد هاي بسياروسيعي در تجارت الکترونيك دارد و شيوه هاي متنوعي را براي فروش محصولات و ارائه خدمات در اختيار تاجران قرار مي دهد

نتايج

- امضاي ديجيتال، الگوريتم ها و سيستم هاي آن در تئوري به خوبي مطرح شده اند.
- در عمل چندان كار آمد نيستند و يا به عبارتي به خوبي در عمل مورد استفاده قرار نمي گيرند.
- پياده سازي و استفاده از برنامه هاي كاربري كه از امضاي ديجيتال استفاده مي كنند كند مي باشد.
- استاندارد DSS توسط دولت آمريكا تاييد شده است و انتظار آن مي رود كه در آينده رقيب خوبي براي RSA باشد.

منابع

- Federal Information Processing Standard (FIPS) Publication 186-2 *Digital Signature Standard (DSS).* US/DoC NIST. January 27, 2000 with Change Notice#1 October 5, 2001.
- National Institute of Standards and Technology (NIST), Secure Hash Standard, FIPS PUB 180-1,
- URL: www.itl.nist.gov/fipspubs/fip180-1.htm
- American Bar Association, Section of Science and Technology, Information Security Committee. "Digital Signature Guidelines Tutorial", 22 June 2003.
- URL:http://www.s2.chalmers.se/iths/pdf/Digital%20signature%20tutorial.pdf FACT SHEET ON DIGITAL SIGNATURE STANDARD, May 1994
- .URL:http://security.isu.edu/pdf/dss_fact.pdf
- Bengisu Tulu, Haiqing Li, Samir Chatterjee, Brian N. Hilton, Deborah Lafky and Thomas A. Horan, Design and Implementation of a Digital Signature Solution for a Healthcare Enterprise, 2004
- URL:http://ncl.cgu.edu/publications/tulu_li_chatterjee_hilton_lafky_horan.pdf
- P. Kitsos, N. Sklavos and O. Koufopavlou, *AN EFFICIENT IMPLEMENTATION OF THE DIGITAL SIGNATURE ALGORITHM*
- URL: http://www.vlsi.ee.upatras.gr/~pkitsos/Kitsos_ICECS02.pdf.
- Internet WWW page at
- URL:http://islab.oregonstate.edu/koc/ece575/03Project/Prabhakararao-Mathur/DSA.html

سئولات