جامعة المنارة الخاصة
كلية الهندسة
هندسة الروبوت والأنظمة الذكية

مهارات الحاسوب

Computer Skills UNRC101

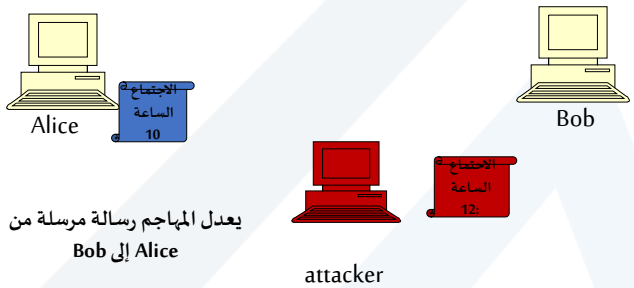
مدرس المقرر
أ.د. مثنى علي القبيلي

الفصل الدراسي الأول | الاثنين 09/01/2023 | العام الدراسي 2022-2023

تابع البعثرة Hash Function

مقدمة

➢ تعرف تكاملية المعطيات على أنها الخاصية التي تسمح بالتحقق من أن المعطيات لم تعدل من قبل كيان غير مخول له بذلك سواء بشكل مفاجئ أو مقصود.



➢ يستخدم تابع البعثرة (hash) عملياً من أجل التحقق من متطلب تكاملية المعطيات.

تعريف تابع البعثرة (Hash function)

➢ **تعريفه:**

هو تابع يربط قناة ثنائية ذات طول متغير مع قناة ذات طول ثابت.

رياضياً هو تابع حسابي فعال يقوم بتحويل السلاسل الثنائية ذات الأطوال المتغيرة إلى سلاسل ثنائية ذات أطوال ثابتة (n). نرمز له بـ h():

$$h:\{0,1\}^{*} \rightarrow \{0,1\}^{n}, m \rightarrow h(m)$$

✓ مثالياً تكون قيم n ما بين 128-256 bits

✓ تدعى قيمة خرج تابع البعثرة بموجز الرسالة (message digest)

✓ يطلق عليه أحياناً : تابع البصمة (fingerprint function)

خصائص تابع الـ Hash (1/4)

❖ يجب أن يقدم تابع الـhash :

1. الضغط (Compression): يجب أن ينتج طول خرج ثابت و أصغر مقارنة مع أطوال الدخل.

Message	*Message Digest*
4523AB1352CDEF45126	13AB
723BAE38F2AB3457AC	02CA
AB45CD1048765412AAAB6662BE	A38B

2. الفعالية(Efficiency): سهل الحساب مهما كان الدخل
من أجل رسالة معلومة m : يكون من السهل حساب h(m)

3. وحيد الاتجاه (One-way): من أجل قيمة y معطاة من غير الممكن إيجاد x بحيث $h(x) = y$

خصائص تابع الـ Hash(2/4)

4. مقاوم للتصادمات (Strong collision resistance): من أجل أية قيمتين مختلفتين ينتج حكماً خرجين مختلفين : $y \neq x \rightarrow h(y) \neq h(x)$

❖ مثال 1: إذا كانت Data $X = (X_0,X_1,X_2,\ldots,X_{n-1})$, each X_i is a byte

إذا فرضنا تابع hash يعرف كالآتي: $h(X) = X_0+X_1+X_2+\ldots+X_{n-1}$

هل هو آمن ؟

❖ الحل:

إذا كان لدينا $X = (10101010,00001111)$

سيكون : $h(X)=10111001$

لكن بفرض $Y = (00001111,10101010)$

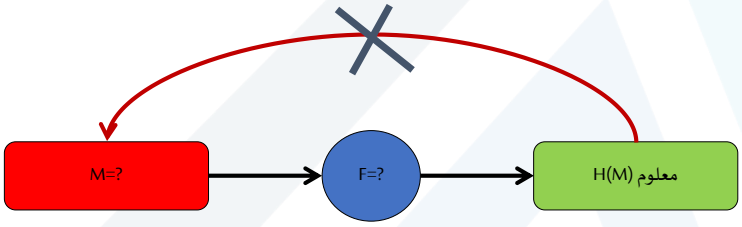
سيكون : $h(Y)=10111001$

من السهل إيجاد دخلين مختلفين لهما نفس قيمة الخرج ، إذاً هذا التابع غير آمن.

خصائص تابع الـ Hash(3/4)

5. مقاوم ضد هجوم الصورة الأولية (Pre-image Attack)

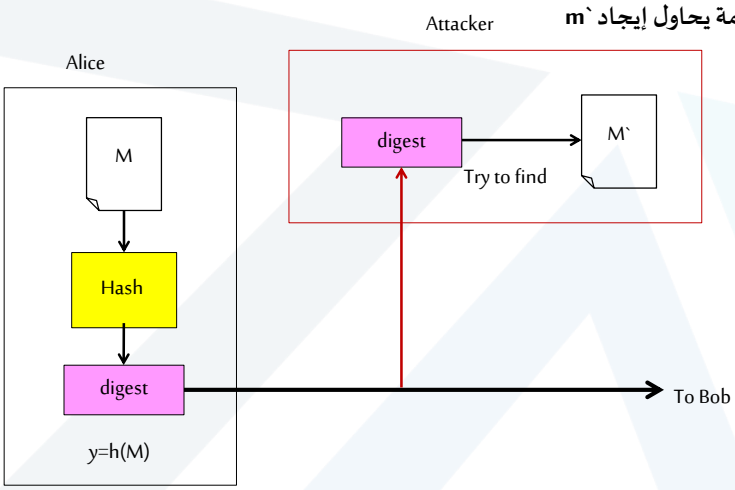
- إذا كانت قيمة التابع h(m) معلومة فإنه من الصعب حساب m



هذه الخاصية تحمي ضد المهاجم الذي يمتلك قيمة تابع البعثرة ويحاول إيجاد قيمة الدخل

Preimage Attack

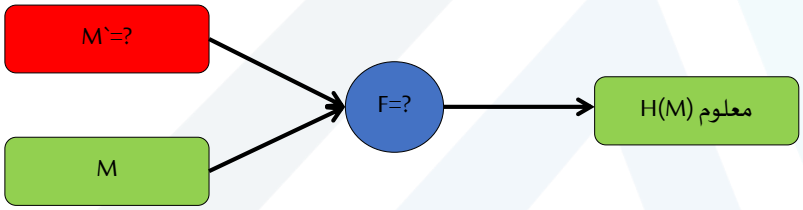
h(m) معلومة يحاول إيجاد `m



خصائص تابع الـ Hash(4/4)

6. مقاوم ضد هجوم الصورة الثانية (Second pre-image Attack)

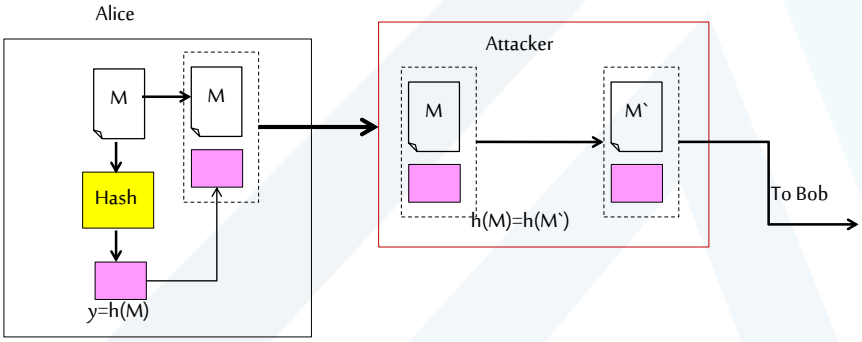
- من أجل قيمة رسالة معروفة m , فإنه من الصعب إيجاد رسالة أخرى **m`** بحيث يكون لها نفس قيمة تابع الـhash أي : hash(m)=hash(m`)



هذه الخاصية تحمي من المهاجم الذي يمتلك الدخل وقيمة خرج تابع البعثرة الموافق له ويريد أن يستبدل القيمة الأصلية بقيمة مختلفة كقيمة شرعية

Pre-image Attack Second

m, h(m) معلومة يحاول إيجاد m` مختلفة عن m
بحيث h(m)=h(m`)



أهم استخدامات تابع الـ Hash

1. تحقيق مصادقة أصل المعطيات
2. تحقيق متطلب المصادقة
3. تحقيق متطلب التكاملية

المصادقة باستخدام تابع البعثرة (1/2)

- في حال أراد طرفان التأكد من هوية كل منهما والتأكد من المفتاح المشترك بينهما
- يقوم أحدهما بإرسال رقم عشوائي هذا الرقم العشوائي يكون بمثابة تحدي (challenge)
- الطرف الذي استقبل التحدي يطبق تابع البعثرة على قيمة هي لصق لهذا التحدي مع المفتاح المشترك، ويقوم بإرسال خرج تابع البعثرة إلى الطرف الآخر .
- عندما يستقبل ذلك الطرف خرج تابع البعثرة يقوم بإعادة الحساب باستخدام الرقم العشوائي والمفتاح، ويقارن النتيجة مع ما تم استقباله.
- هنا نميز حالتين:
  - ✓ في حال التساوي تكون المصادقة محققة
  - ✓ في حال عدم التساوي تكون المصادقة غير محققة

المصادقة باستخدام تابع البعثرة (2/2)

تكاملية البيانات باستخدام تابع البعثرة (1/2)

- إن استخدام تابع البعثرة هنا يفيد في التأكد أن الرسالة الأصلية لم يتم تعديلها ، ولكن لا يضمن التحقق من أصل المرسل

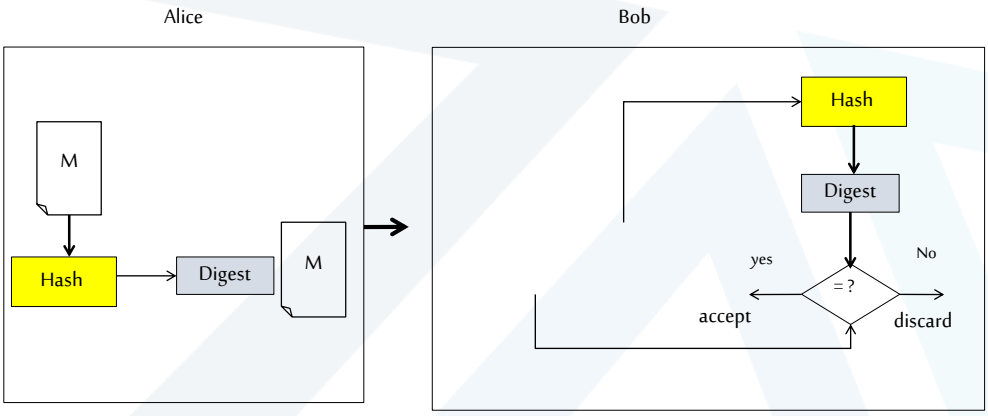
- يقوم المرسل بتطبيق تابع البعثرة على الرسالة و من ثم إرسال ناتج البعثرة (digest) ملصق مع الرسالة

- في جهة الاستقبال يحسب المستقبل تابع البعثرة للرسالة الواصلة إليه ويقارن الناتج مع قيمة تابع البعثرة المستقبلة

❖ في حال التطابق الرسالة صحيحة لم تتعرض للتعديل ومتطلب التكاملية محقق

❖ في حال عدم التطابق الرسالة معدلة و متطلب التكاملية غير محقق

تكاملية البيانات باستخدام تابع البعثرة (2/2)



تصنيف توابع البعثرة (1/2)

تصنف إلى نوعين أساسيين:

❖ <u>توابع بعثرة دون مفتاح :</u>

هي توابع البعثرة التي تمتلك دخل واحد هو الرسالة .

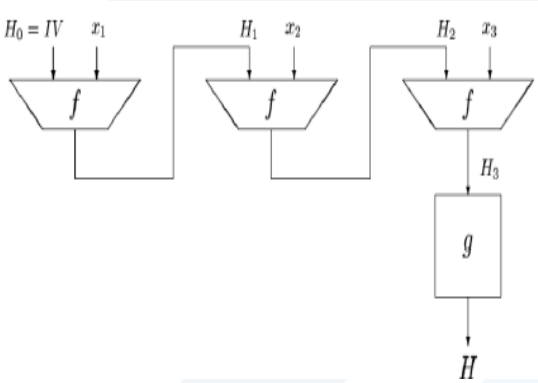
مثال عنها **MDC (Manipulation Detection Code)**

- في MDC : خرج تابع البعثرة يلصق مع المعلومات الأصلية التي تم تطبيق التابع عليها ومن ثم تشفيرها بهدف الحماية من العبث فيها. أي يُرسل في قناة موثوقة كالآتي:

$$MDC = E_K(M \| H(M))$$

حيث K هو المفتاح السري.

- أغلب توابع الـ MDC مبنية على تكرار تابع البعثرة وتقسيم الرسالة إلى بلوكات



$$
\begin{aligned}
h_0 &= IV = \text{a fixed initial value} \\
h_1 &= H(h_0, M_1) \\
&\vdots \\
h_i &= H(h_{i-1}, M_i) \\
h_n &= H(h_{n-1}, M_n) \\
h(M) &= h_n
\end{aligned}
$$

حيث: F هو تابع البعثرة. g هو تابع تحويل.

تصنيف توابع البعثرة (2/2)

تصنف إلى نوعين أساسيين:

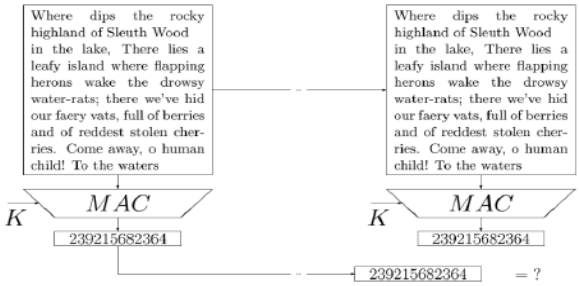
❖ توابع بعثرة مع مفتاح :

هي توابع البعثرة التي أحد مداخلها هو المفتاح السري و المدخل الآخر هو الرسالة.

مثال عنها MAC(Message Authentication Code)

يتم فيها حساب MAC و إرسال:

$$M \| MAC_K(M)$$

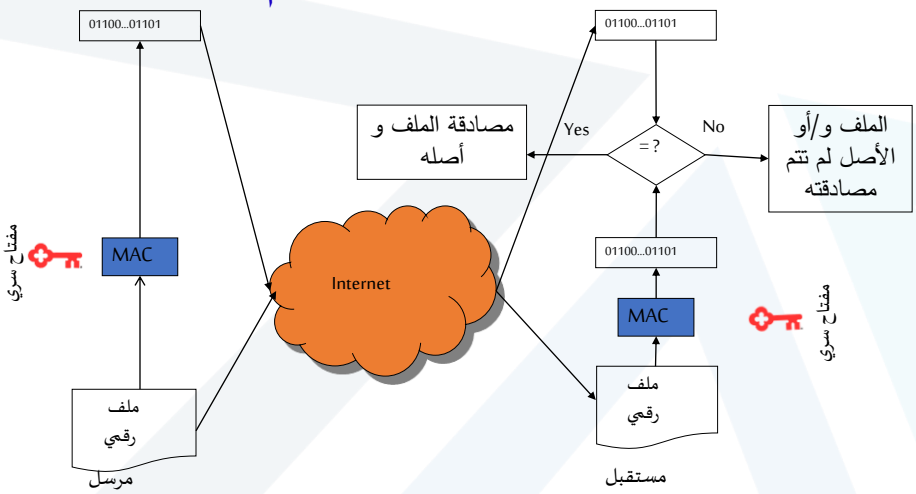


Message Authentication Code (MAC)

➢ آلية العمل:

- ✓ يتشارك المرسل A والمستقبل B بنفس المفتاح السري
- ✓ يحسب المرسل $MAC_{K_{AB}}(m)$ و تلصق مع الرسالة (m) من أجل إرسالها باستخدام المفتاح السري K_{AB}
- ✓ يرسل المرسل $m \| MAC_{K_{AB}}(m)$
- ✓ بعد استقبال الرسالة , يبحث المستقبل عن مصدر الرسالة المستقبلة كما يلي:
  - يعيد المستقبل حساب الـ$MAC_{K_{AB}}(m)$ للرسالة المستقبلة باستخدام المفتاح السري K_{AB}
  - يقارن هذه النتيجة مع الـ $MAC_{K_{AB}}(m)$ المستقبلة ويرى إن كانت الرسالة هي نفسها المستقبلة و أصلية مرسلة من المصدر أو لا

مصادقة أصل المعطيات باستخدام الـ MAC



أمثلة عن Hash Function

- MD2 (Message Digest 2)
- MD5 (Message Digest 5)
- SHA-1 (Secure Hash Algorithm 1)

سرعة الحساب (ميغابايت/ثانية)	Digest Size (بت)	Hash Algorithm
204.55	128	MD5
72.60	160	SHA-1

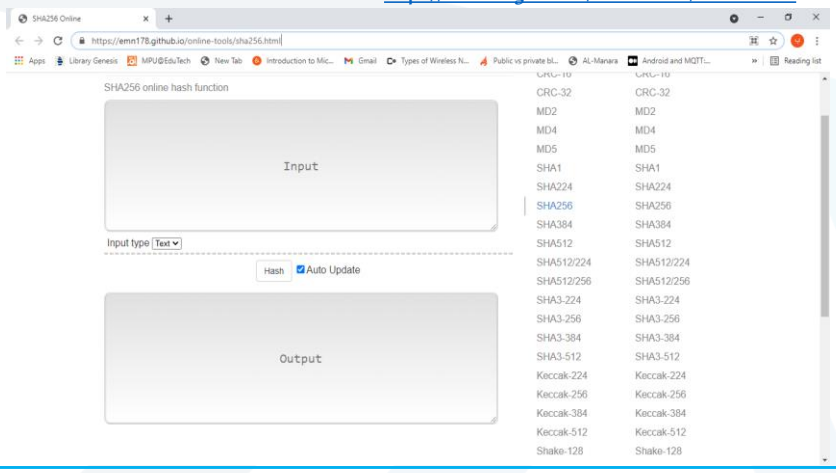
على جهاز مع معالج Pentium 4، 2.1 GHz وتحت Win XP SP1

- مثال عن الـMAC هو الخوارزمية HMAC/MD5

سرعة الحساب (ميغابايت/ثانية)	Digest Size (بت)	MAC Algorithm
215.76	128	HMAC/MD5

تدريب على تابع البعثرة

https://emn178.github.io/online-tools/sha256.html



تدريب على تابع البعثرة

SHA1 online hash function

HELLO

Input type Text

Hash Auto Update

c65f99f8c5376adadddc46d5cbcf5762f9e55eb7

SHA1 online hash function

HALLO

Input type Text

Hash Auto Update

163f5910b94225c12e07390756946d4b1794f5d1

تدريب على تابع البعثرة

SHA1 online hash function

HELLO

Input type Text

Hash Auto Update

c65f99f8c5376adadddc46d5cbcf5762f9e55eb7

SHA1

SHA1 online hash function

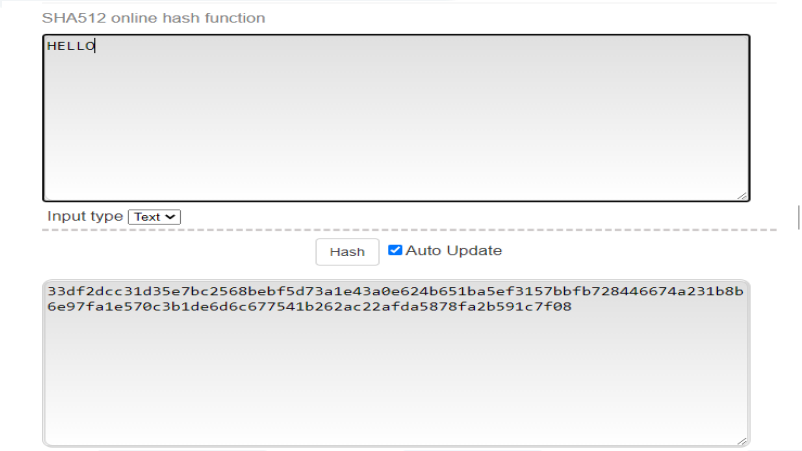
HELLO HOW ARE YOU

Input type Text

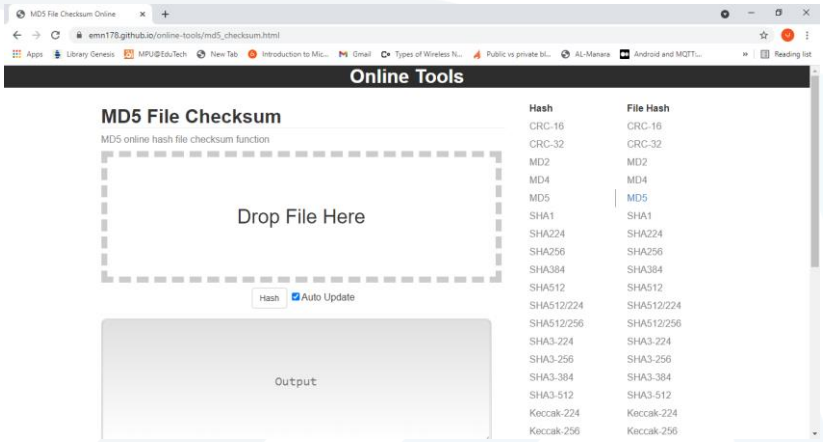
Hash Auto Update

d387e630d3fd8f69865a2928f785abab93177d0e

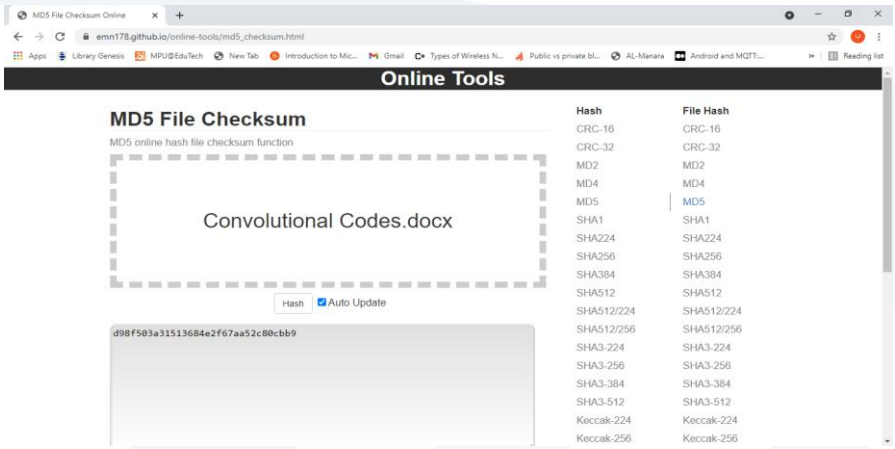
تدريب على تابع البعثرة



تدريب على تابع البعثرة



تدريب على تابع البعثرة



التوقيع الرقمي (Digital Signature)

تعريف التوقيع الرقمي Digital Signature

- يضمن التوقيع الرقمي عدم التنصل للأصل (Non-repudiation to origin) أي أن المرسل لن يكون مستقبلاً قادراً على أن يتنصل من أنه هو من أرسل الرسالة.
- هو آلية تقوم على نظام تشفير غير متناظر
- يحسب التوقيع الرقمي باستخدام المفتاح الخاص للمرسل و يتم التحقق منه بواسطة المفتاح العام للمرسل.

تمثيل التوقيع الرقمي رياضياً

1. يوقع بوب الرسالة باستخدام مفتاحه الخاص اعتماداً على خوارزمية التوقيع المستخدمة:

$$S = E_{K_{priB}}(M)$$

2. يرسل بوب الرسالة و التوقيع الرقمي معاً:

$$S \| M$$

3. تستقبل أليس الرسالة والتوقيع

4. تقوم بالتحقق من التوقيع كالآتي:

$$V_{K_{pubA}}[S(M)]$$

1-تحسب الرسالة اعتماداً على التوقيع المستقبل:

$$M_{cal.} = D_{pubB}(S) = D_{pubB}\left(E_{K_{priB}}(M)\right)$$

2- تقارن الرسالة المحسوبة مع المستقبلة:

$$M_{cal.} ? = M$$

يكون التوقيع فعال وصحيح في حال التساوي

$$M_{cal.} = M$$

التوقيع الرقمي مع الختم الزمني

إرسال الرسالة مع التوقيع

↓

إمكانية إعادة استخدامهما أكثر من مرة من قبل المستقبل

↓

الخطورة تكمن في حال الشيكات الرقمية

↓

يمكن للمستقبل الاستفادة من الشيك و سحب المبلغ أكثر من مرة

لذا يتضمن التوقيع الرقمي ختماً زمنياً **Timestamp** :

- تاريخ التوقيع
- زمن فعاليته

توقع هذه المعلومات مع الرسالة

التوقيع الرقمي مع تابع البعثرة (Hash)

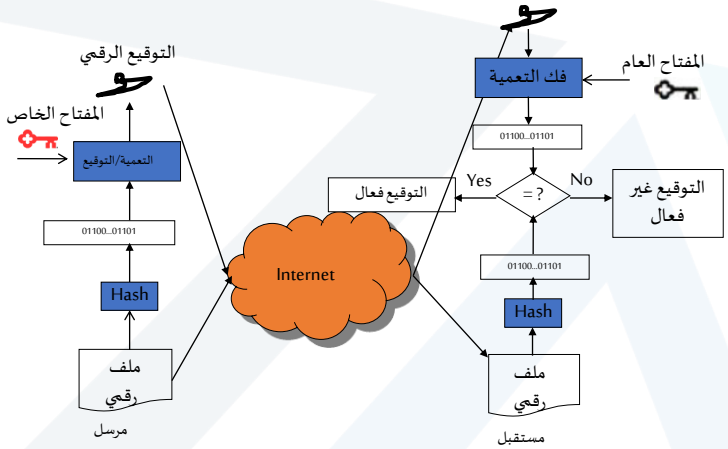
يستغرق التوقيع الرقمي وقتاً طويلاً للحساب في حال الرسائل الطويلة

لذلك

↓

استخدام التوقيع الرقمي على خرج تابع الـHash

عدم التنصل للأصل باستخدام التوقيع الرقمي



استخدام التوقيع الرقمي مع التشفير

Alice | **Bob**

$$S = E_{K_{pri_A}}(M)$$

$$E_{K_{pub_B}}[S(M)] \longrightarrow D_{K_{pri_B}}\left[E_{K_{pub_B}}[S(M)]\right] = S(M)$$

$$V_{K_{pub_A}}[S(M)] = M$$

من أجل التوقيع الرقمي (S):

✓ يولد المرسل التوقيع الرقمي (S) انطلاقاً من الرسالة M باستخدام مفتاحه الخاص :

$$S = M^{d} \bmod(n)$$

حيث أن المفتاح الخاص هو: $K_{pri} = \{d, n\}$

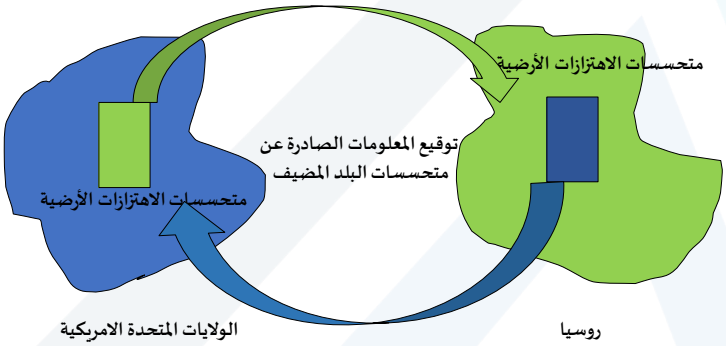
✓ يستقبل المستقبل الرسالة M و التوقيع الرقمي (S) ويتحقق من الرسالة M باستخدام المفتاح العام للمرسل :

$$M = S^{e} \bmod(n)$$

حيث أن المفتاح العام هو : $K_{pub} = \{e, n\}$

من تطبيقات التوقيع الرقمي

التحقق من معاهدات حظر التجارب النووية



من تطبيقات التوقيع الرقمي

البطاقة المصرفية

❖ إن شريحة البطاقة المصرفية هي عبارة عن حاسوب صغير, يحتوي على CPU,RAM,ROM ....إلخ.

❖ من أجل مصادقة البطاقة، يخزن في البطاقة:

✓ قيمة معرِّف (محدد)(VI): تكوَّن هذه القيمة من معلومات تخص البطاقة:

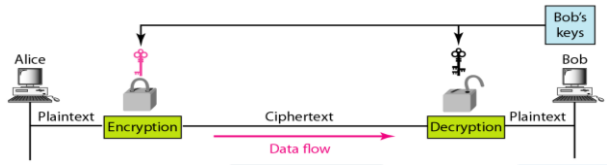
مثل (حامل البطاقة, عدد, تاريخ الفعالية, الشعار)

✓ قيمة المصادقة (VA): و هي قيمة VI المشفرة باستخدام الخوارزمية RSA باستخدام المفتاح الخاص للموزع (GIE)

❖ عند وضع البطاقة في الطرفية، يتم التحقق من أن القيمة الناتجة من فك تشفير VA باستخدام المفتاح العام للموزع مطابقة لقيمة VI المخزنة ضمن البطاقة

الفرق بين نظام التشفير والتوقيع الرقمي من حيث المفاتيح

✓ **في نظام التعمية** : يستخدم المفتاح العام و المفتاح الخاص للمستقبل .



✓ **في التوقيع الرقمي:** يستخدم المفتاح الخاص و المفتاح العام للمرسل.

Thanks