CSC3631 Cryptography
- Digital Signatures II

Thomas Gross

What are Characteristics of Digital Signatures?

Goal?

Who can verify?

Whom can one show a signature?

Deniability?

What's the security property?

Existential Unforgeability I

No adversary should be able to forge any signature.

Setup: Generate keypair (pk, sk)

Inputs to Adversary **A:** pk, access to $\mathbf{Sign}_{sk}()$

A gets signatures on an arbitrary set of messages m in Q.

Output by **A**: message-signature pair (m^*, σ)

Success criterion for **A:** $\mathbf{verify}_{pk}(m^*, \sigma) = \mathbf{1}$

m^* not in Q.

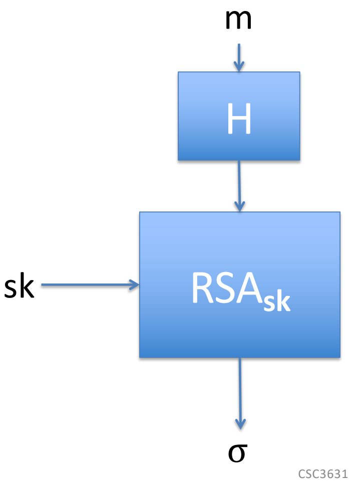
Roadmap

- **RSA Signatures**
  - **Hash&Sign Paradigm**
- Digital Signature Standard (DSS)
- Certification Infrastructures

Goal for today:
- **How is the RSA and DSS Signature Schemes realized?**
- **How do we reach of full certification infrastructure?**

Hashed RSA

How to get an actual signature scheme out of RSA?



Hashed RSA can be proven is **existentially unforgeable under an adaptive chosen-message attack** under the RSA assumption and in the **random oracle model**.

Roadmap

- RSA Signatures
  - Hash&Sign Paradigm
- **Digital Signature Standard (DSS)**
- Certification Infrastructures

Goal for today:

- **How is the RSA and DSS Signature Schemes realized?**
- **How do we reach of full certification infrastructure?**

DSS Key Generation

How to create a strong setting for DSS?

GenDSS(1^n)

Input: key length n

Create a cyclic group G, sub-group of $(\mathbf{Z}_p)^*$
with generator g with prime-order q.
q divides p-1, but q^2 does not divide p-1

Choose random x in $\mathbf{Z}_q$

Compute $y = g^x$ (**mod** p)

Output: $pk=(G, q, g, y)$, $sk=(G, q, g, x)$

Structure of the signature

Let's have a closer look (first)

Secret key: x **Randomness:** $r = g^k$

$$s = (\mathbf{H}(m) \;+\; xr) \;\cdot\; k^{-1} \quad (\mathbf{mod}\ q)$$

Note: we are working in $\mathbf{Z}_q$

Digital Signature Standard

KeyGen: $pk=(p, q, g, y)$, $sk=(p, q, g, x) \leftarrow$ **GenDSS**(1^n)

Sign: Given $sk=(p, q, g, x)$ and message m:

Choose k random in $\mathbf{Z}_q$; $r = g^k$ (**mod** p)(**mod** q)

σ: r, $s = k^{-1}(\mathbf{H}(m) + xr)$ (**mod** q)

Verify: Given $pk=(p, q, g, y)$, m and signature σ:

Compute $u_1 = \mathrm{H}(m) \cdot s^{-1}$ (**mod** q) and

$u_2 = r \cdot s^{-1}$ (**mod** q)

Check $r = g^{u_1} y^{u_2}$ (**mod** p) (**mod** q)

Correctness of DSS

We call $\mathbf{m} := \mathbf{H}(m)$. **We have** $y = g^x$

$r = g^k$ (**mod** p) (**mod** q)

$s = (\mathbf{m} + xr) \cdot k^{-1}$ (**mod** q)

Working in the main group:

$g^{\mathbf{m}s^{-1}} y^{rs^{-1}} = g^{\mathbf{m}s^{-1}} (g^x)^{rs^{-1}} = g^{(\mathbf{m}\, s^{-1}) + (xr\, s^{-1})}$ **| all** (**mod** p)

We can make our lives easier… (with Euler’s Theorem)

“Working in the exponent” $\mathbf{Z}_q$: **| all** (**mod** q)

$(\mathbf{m}\, s^{-1}) + (xr\, s^{-1}) = (\mathbf{m} + xr)\, s^{-1}$

$= (\mathbf{m} + xr)\, ((\mathbf{m} + xr) \cdot k^{-1})^{-1} = k$

Summary DSS

- The Digital Signature Standard is an **international standard** (proposed by NIST),
- widely used in practice.
- It has been scrutinized for years w/o any attack being found.
- **No security proof** exists.

Roadmap

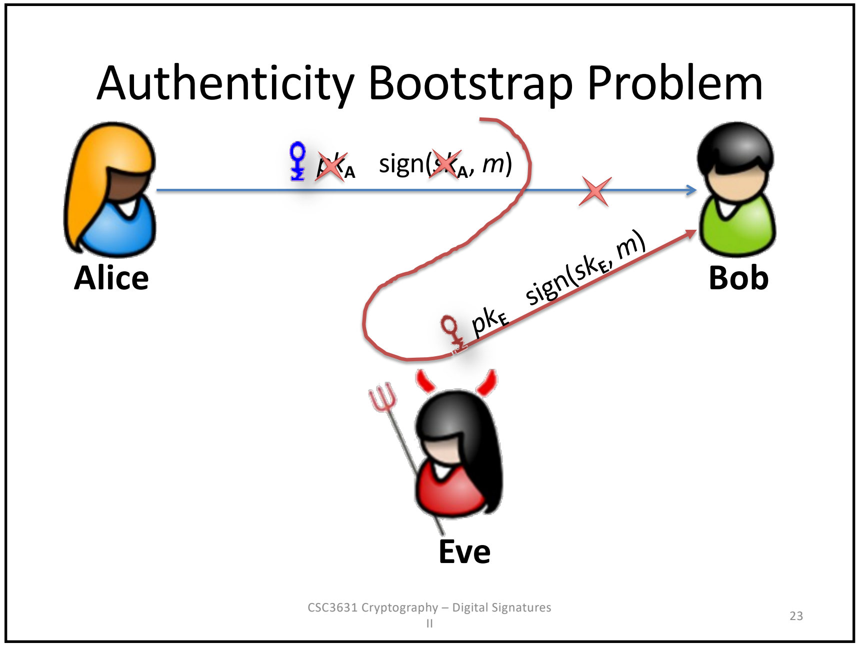
- RSA Signatures
  - Hash&Sign Paradigm
- Digital Signature Standard (DSS)
- **Certification Infrastructures**

Goal for today:

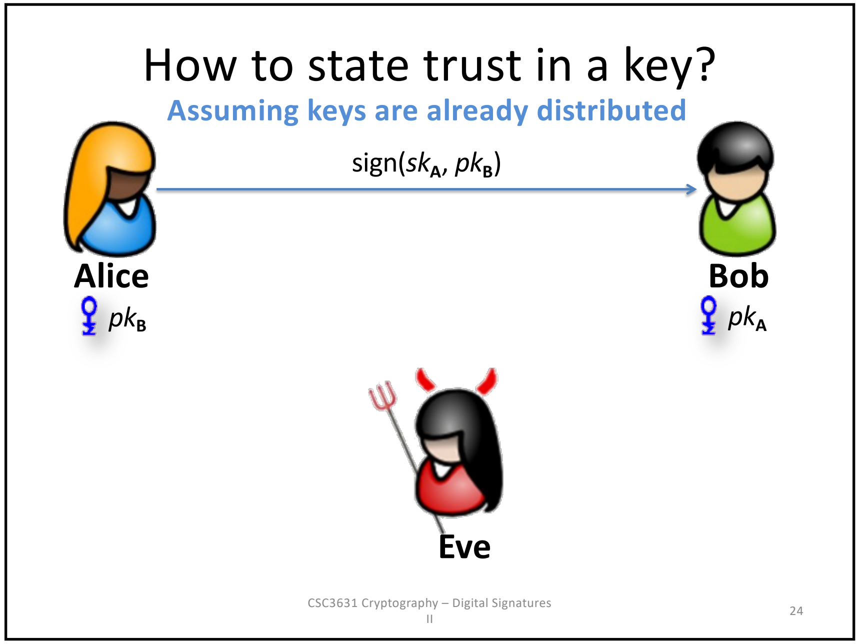
- **How is the RSA and DSS Signature Schemes realized?**
- **How do we reach of full certification infrastructure?**


How to verify that a pk is authentic?
PGP Public Key
• The key has NO legal relevancy. I do not accept legal responsibility for messages signed with respect to it.
• Still, I like getting encrypted messages.
• Obtaining the public key via the Internet is not a secure way. To convince yourself of the key's integrity, please
Key-ID: 0x10AE74D2
Fingerprint: 50B8 9A90 674D 03C4 0C2B B849 2044 CDFF 10AE 74D2
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: PGPfreeware 6.5.1 Int. for non-commercial use
mQGiBDh12XcRBADtgG+6YrlY/n2WfMwTTeQi/lppLgmDGMFweea3fi3+N30NJfeM
wijDWY7LE+giedsEpm6e9CYKUbNmgJhF+icgxd9pWFlAqSFXWlSPsWI2dYxEbgSX
uJohkg86ybF6BYOsKjiPmWO+fY4oWhaGsoPD4H7DGanPDSOUnnj9MVo6PQCg/5RC
fuoKI7TBix9aNA6A1C8WlXMEAOdRCFTX3XKptS9DUXGQZ4CdBbzYO9TpUF+GMWdb
YBT3uY2uML3wTEIAu5Jth1pshP4Y9GcLxI/hUh17ZREWxqV3KLS03QKP0xWbZUo6
CSC3631 Cryptography – Digital Signatures
II
22

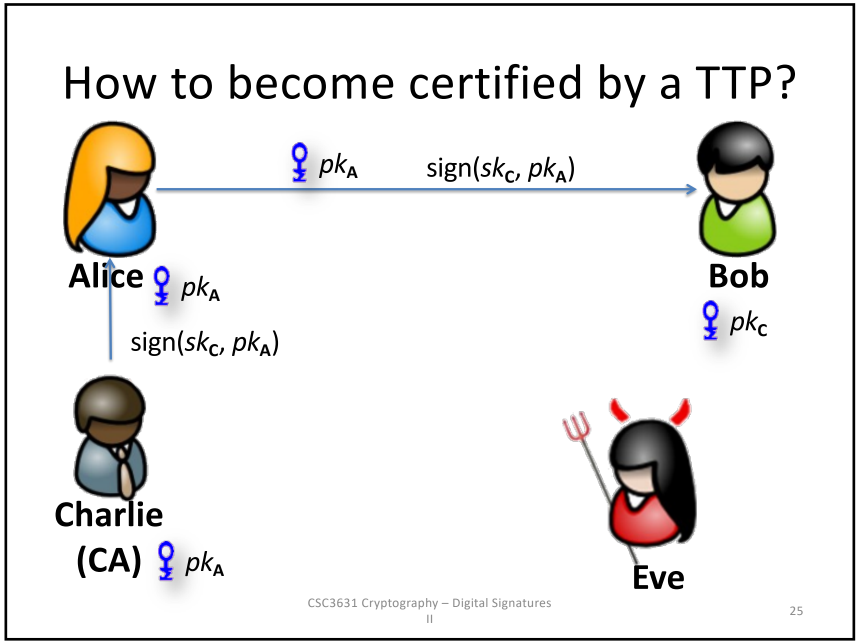
22


Authenticity Bootstrap Problem
Alice
pk_A
sign(sk_A, m)
Bob
pk_E
sign(sk_E, m)
Eve
CSC3631 Cryptography – Digital Signatures
II
23

23


How to state trust in a key?
Assuming keys are already distributed
sign(sk_A, pk_B)
Alice
pk_B
Bob
pk_A
Eve
CSC3631 Cryptography – Digital Signatures
II
24

24


How to become certified by a TTP?
pk_A
sign(sk_C, pk_A)
Alice
pk_A
Bob
pk_C
sign(sk_C, pk_A)
Charlie
(CA)
pk_A
Eve
CSC3631 Cryptography – Digital Signatures
II
25

25

Public Key Certificates

- A *public-key certificate* is a data structure consisting of a *data part* and a *signature part*.
- The *data part* contains cleartext data including, as a minimum, a public key and a string identifying the *subject entity* to be associated with it.
- The *signature part* consists of the digital signature of a certification authority over the data part.
- It, thereby, binds the subject entity's identity to the specified public key.

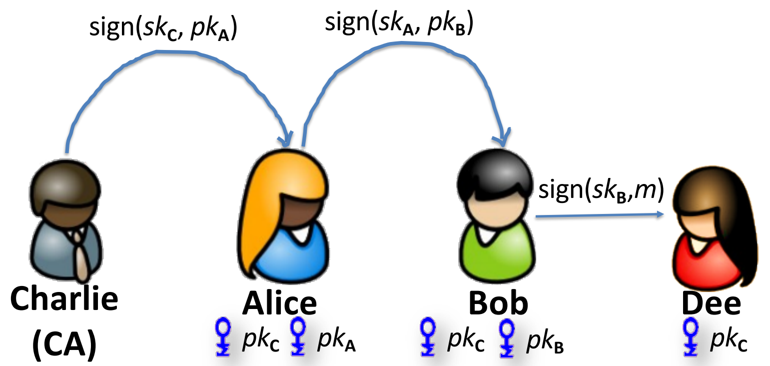
Auxiliary Data in Certificates

- Validity period of the public key
- A serial number/key identifier identifying the certificate/key
- Additional information about subject entity
- Additional information about key (e.g., algorithm, intended use)
- Quality measures related to identification, generation of key pair, etc.
- Information facilitating the verification of the signature

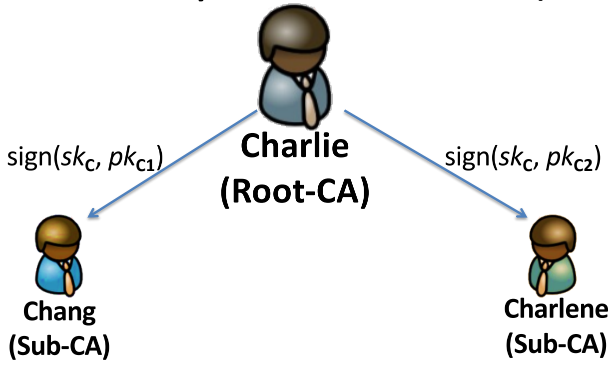
Certificate Verification Procedure

1. Acquire the authentic public key pk_C of the CA
2. Obtain an identifying string id_A which uniquely identifies party **A**
3. Acquire over an unsecure channel the public-key certificate pk_A of party **A**, agreeing with the identifying string id_A.
4. Verify:
   a) Current date and time against the validity period of pk_A
   b) Current validity of CA's public key pk_C
   c) Signature om **A**'s certificate using the CA's pk_C
   d) Certificate on pk_A not revoked
5. If all checks succeed, accept pk_A in the certificate as authentic public key.

How to create a certificate chain?

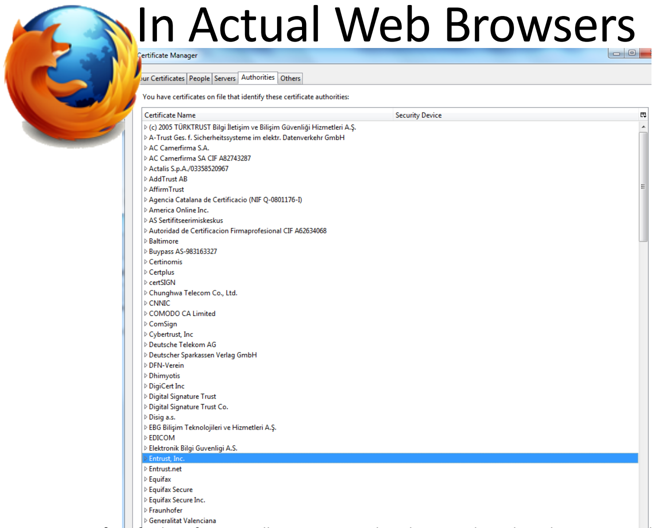


Public Key Infrastructure (PKI)



30

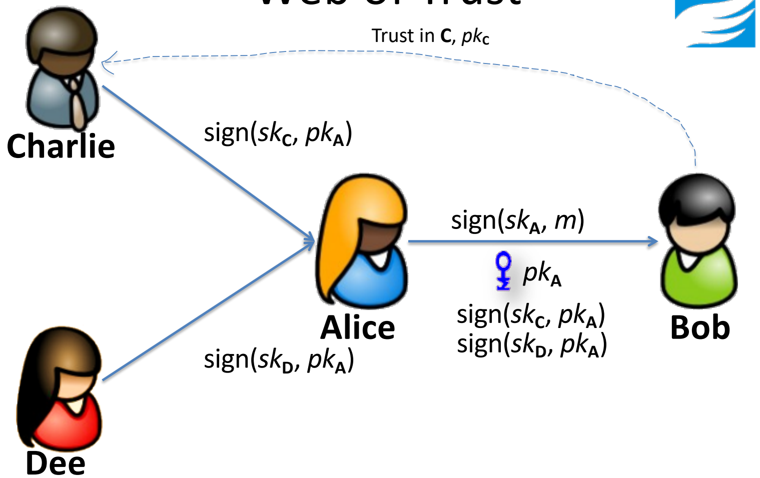
In Actual Web Browsers



[Firefox logo from Mozilla.org, as example only. No trademark implications intended.]

31

Web of Trust



[GPG logo from gnupg.org, as example only. No trademark implications intended.]

32

Summary

All public-key crypto depends on **authentic key distribution**.

Public-key signatures serve as **bootstrap mechanism** for the distribution.

Certificates and certification chains establish and delegate trust.

33