The slides are loosely based on those of Prof. Mihir Bellare, UC San Diego.
Agenda

1. High-level Overview

2. Building Signature Scheme
The Need For Signing Is Ubiquitous
How To Sign Electronically?

Problem: A digitized signature is easily copied \(\rightarrow\) forgery

Lots of apps to digitize signatures
Digital Signature Scheme: Syntax

Key Gen

\[ \mathcal{K} \rightarrow \$ \rightarrow \text{pk} \rightarrow \text{sk} \]

Sign

\[ M \rightarrow \text{Sign} \rightarrow \$ \rightarrow \sigma \rightarrow \text{sk} \]

Verify

\[ \sigma \rightarrow M \rightarrow \text{Ver} \rightarrow \text{pk} \rightarrow 0/1 \]
# Digital Signature versus MAC

<table>
<thead>
<tr>
<th>MAC</th>
<th>Digital Signature</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Verifier needs to share a secret key with signer</td>
<td>- Verifier needs no secret</td>
</tr>
<tr>
<td>- Verifier can impersonate signer</td>
<td>- Verifier cannot impersonate signer</td>
</tr>
</tbody>
</table>
Digital Signature: Unforgeability Security

- Similar to MAC security
- **Difference**: The adversary is given the public key

Again, digital signature doesn’t directly thwart replay attack.
Agenda

1. High-level Overview

2. Building Signature Scheme
A Bad Scheme: Plain RSA Signature

Key generation: Like RSA encryption

Sign:
- To sign a message, “decrypt” it:

Verify:
- To verify a signature, “encrypt” it and compare with the message
Issues with Plain RSA Signature

- **Feasibility**: Can sign only short messages
- **Security**: Can easily break unforgeability security

No sign query needed!

\[ pk = (n, e) \]

\[ (M', \sigma') = (x^e \mod n, x) \]
Exercise: Forging Plain RSA For Targeted Msg

**Goal:** The forged message must be a specific one

\[ pk \rightarrow (M' = M_1 \cdot M_2, \sigma') \]
Hash-then-Sign Paradigm

Plain RSA Signature → Full Domain Hash (FDH)

**Key generation**: Like Plain RSA

**Sign**: To sign message $M$

![Diagram]

**Question**: How to verify?
Security Requirement for Hash Function

What intuition suggests: Hash must be collision-resistant

If $H(M) = H(M')$ then $M$ and $M'$ have the same signature

What proof requires: Hash is modeled as a random oracle
A Gap of **Demand** and **Supply**

2048

bits of output

\[ M \]

\[ H \]

Plain RSA Signing

512

bits of output

\[ M \]

SHA
A Common Wrong Way to Hash

2048-bit string $Y$, viewed as a number in $\mathbb{Z}_N$

Broken by Desmedt and Odlyzko in 1985
How to Hash Properly

Use the first $m = \lceil \log_2(N) \rceil$ bits and take mod $N$
Hashing in PKCS#1

19 bytes to indicate what hash function and its output length

2 bytes padding

0001 FF FF ... FF 00 hash info

2048-bit string $Y$, viewed as a number in $\mathbb{Z}_N$