

A Brief Introduction to Provable Security

Dr. Alexander W. Dent
Information Security Group
Royal Holloway, University of London



Contents

- An introduction
- Four flavours of provable security
- Games involving signatures
- Proving the impossible
- Arguments about proofs
- Playing tight
- [A tale of two signature schemes]



An Introduction (to the Introduction)

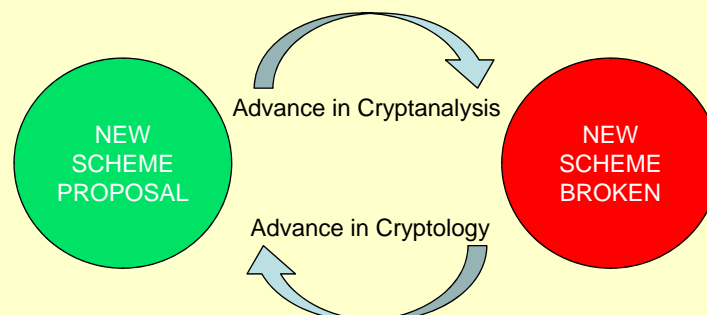
- Evolution vs. Intelligent Design
- What does provable security mean?
- The great complexity theoretic barrier



Royal Holloway
University of London

Evolution of Cryptography

- Traditionally, cryptographic algorithms and protocols were produced by evolution.



Royal Holloway
University of London

Evolution of Cryptography

- Evolution is a poor methodology
 - Too few lions thinning the herd
 - No concept of competitive advantage until an algorithm is deployed
 - The effects of a bad design can be extreme
 - Short periods of time for evolution to occur
 - Schemes often selected by reputation rather than by algorithmic design choices



Royal Holloway
University of London

What is provable security?

- A methodology which seeks to provide reliable assurances that an algorithm or protocol is secure using mathematical techniques.
 - Theorem-Proof paradigm
 - Needs interpretation to apply to the real world



Royal Holloway
University of London

What is provable security?

- We wish to prove that no reasonable attacker can break a scheme in practice.
 - We need a precise definition of the capabilities of a reasonable attacker.
 - We need a precise definition of what an attacker must achieve to break the scheme.
 - We need a mathematical proof which demonstrates that no “reasonable attacker” can achieve this “win condition”.

The Great P=NP Barrier

- Breaking most cryptographic schemes is a computational problem.
- For example, breaking RSA encryption is a computational problem.
- However, if presented with a solution, then it is often easy to check that the solution is correct.

The Great P=NP Barrier

- Cryptography is about problems that are computational hard to solve but computational easy to verify.
- This is the definition of NP problems.
- We cannot show that there problems of this form exist (P=NP problem).
- (Almost) always a computational assumption hiding in the background.



Royal Holloway
University of London

Four Flavours of Provable Security

- Four flavours
- A taste of each flavour



Royal Holloway
University of London

Four Flavours

- There are four main flavours of provable security.
 - These differ in the way in which we define the security model.
 - These divide (roughly) into two classes depending on the proof techniques.



Four Flavours

- The four flavours are:
 - Information-theoretic security
 - Proofs based on formal methods
 - Game-based notions security
 - Simulation-based notions of security
- The first two are combinatorial in nature
- The last two are algorithmic in nature



Formal Methods

- Perhaps I'm not the best person to talk about this subject...
- Mathematical cryptographers believe:
 - It's all about symbolic manipulation
 - It completely abstracts algorithms to ideal assumption about their functionality
 - Mostly use combinatorial and discrete maths techniques to prove the impossibility of breaking a protocol.



Royal Holloway
University of London

Information-Theoretic Security

- Also known as unconditional security
- Does not rely on underlying computational assumptions, but on limited access
- Very difficult to use with public-key cryptography
- Examples include Shannon's one-time pad, Shamir secret sharing, combinatorial key pre-distribution schemes, ...



Royal Holloway
University of London

Game-Based Security

- Security model is phrased as a game played by a resource-bounded attacker
- Typically, we use poly-time attackers
- Sometimes, we use bounded storage too
- The security model defines:
 - the inputs to the attacker
 - the oracles* to which the attacker has access
 - the win condition for the attacker



Game-Based Security

- Security results only apply if scheme is used in a situation which matches the model:
 - the attacker is given no inputs except those available in the model
 - the attacker is given no abilities (oracles/ space/time) except those available in model
 - achieving the “win condition” is the only way to break the scheme



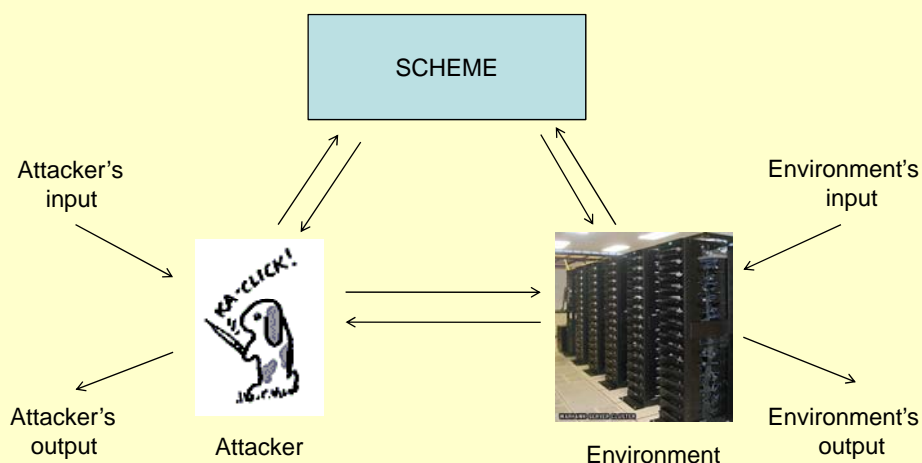
Simulation-Based Security

- Considers the interaction between the scheme, an arbitrary attacker, and an arbitrary environment (representing everything else in the system).
- Attacker and environment often resource constrained (e.g. poly-time algorithms).
- We wish to consider the attacker's output and the environment's output.



Royal Holloway
University of London

Simulation-Based Security



Royal Holloway
University of London

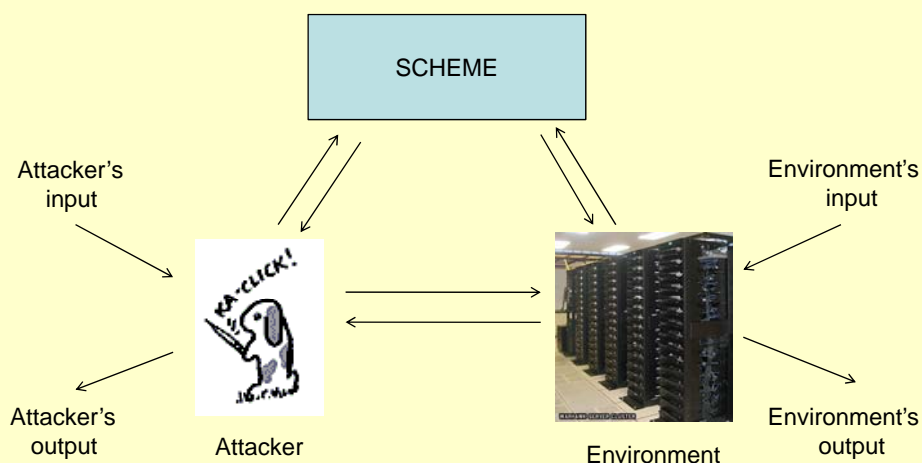
Simulation-Based Security

- We define an ideal version of the algorithm (which is unbreakable by definition)
- The scheme is secure if the outputs of the attacker and the environment are essentially the same when interacting with the real scheme and the ideal scheme.



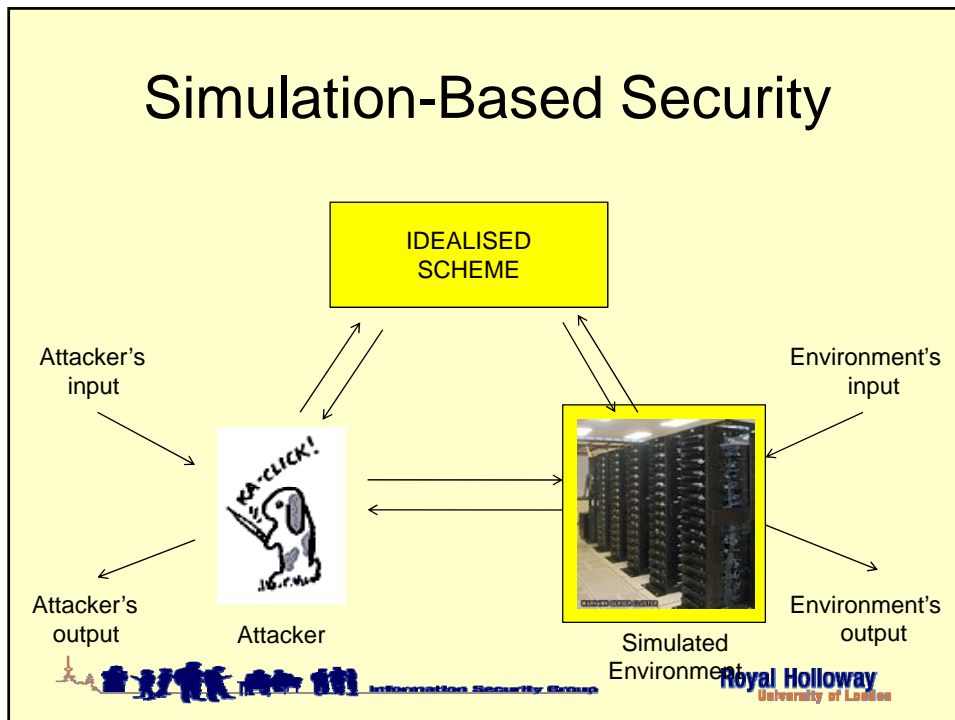
Royal Holloway
University of London

Simulation-Based Security



Royal Holloway
University of London

Simulation-Based Security



Simulation-Based Security

- If the outputs are the same in both situations, then
 - any attack against the real scheme will also work against the idealised scheme;
 - however, no attacks work against the idealised scheme (by definition).
- Scheme can be used in any situation (unlike game-based security).
- Not all schemes can be proven secure.

A Game Involving Signatures

- The game-based security model for digital signatures



Royal Holloway
University of London

Digital Signatures

- A signature scheme is three algorithms:
 - Key generation algorithm: produces public keys pk and private key sk .
 - Signature algorithm: takes a private key and message as input, and produces signatures.
 - Verification algorithm: takes a public key, message and signature as input, and outputs either true or false.
- No attacker can produce signatures.



Royal Holloway
University of London

Digital Signatures

- Attacker would know the public key.
- An unbounded attacker can certainly break the scheme – so what is a “reasonable attacker”?
- It’s likely that the attacker can see some message/signature pairs, but from what message distribution?
- What does it mean to say the attacker breaks the scheme/wins the game?



Royal Holloway
University of London

Digital Signatures

- The security model for digital signatures was given by Goldwasser, Micali and Rivest in 1988.



Royal Holloway
University of London

Digital Signatures

- Attacker is any polynomial-time algorithm.



Royal Holloway
University of London

Digital Signatures

- Attacker is given a randomly generated public verification key.

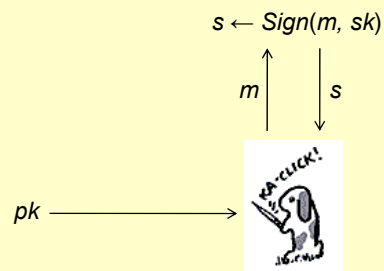
pk →



Royal Holloway
University of London

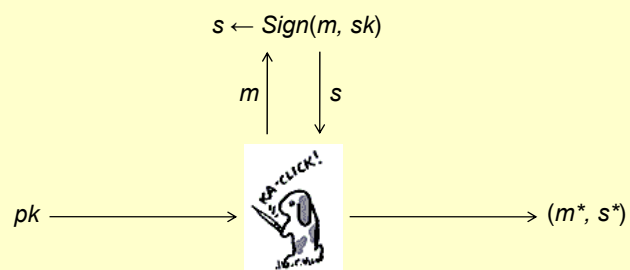
Digital Signatures

- Attacker has access to a signing oracle.



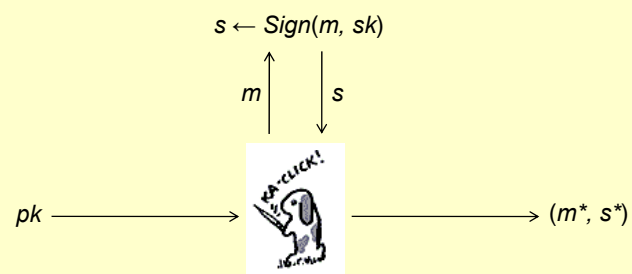
Digital Signatures

- Attacker outputs a message and signature.



Digital Signatures

- Attacker wins if s^* is a valid signature for m^* and the attacker never asked for the signature of m^* .



Proving the Impossible

- Complexity theoretic reductions
- Game hopping
- More sophisticated techniques



How do we prove security?

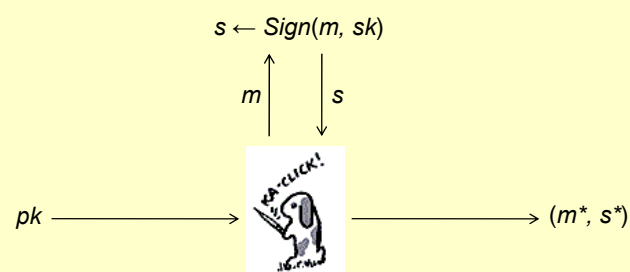
- Security is typically proven using complexity theoretic reductions.
- Similar to the theory of NP-completeness.

“If there exists an attacker that runs in time t and has success probability ε then there exists an algorithm that solves a hard problem in time t' with success probability ε' .”



Royal Holloway
University of London

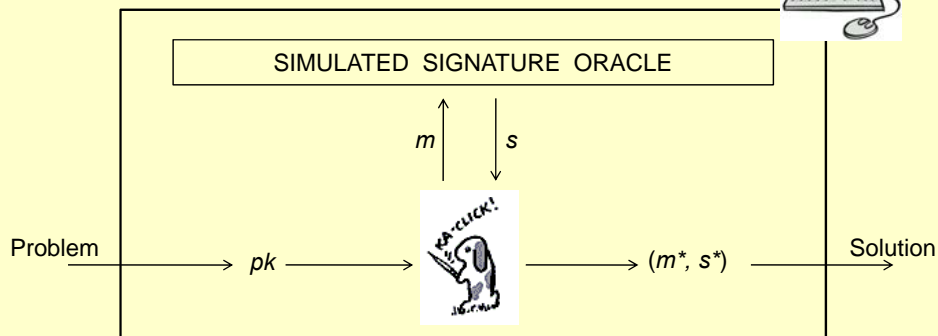
How do we prove security?



Royal Holloway
University of London

How do we prove security?

- The tough part is the simulating the signature oracle



How do we prove security?

- Simple simulations are hard to achieve without using the random oracle model.
- The random oracle model replaces hash functions with random functions which are accessible as an oracle.
- The algorithm can simulate the random oracle in any way as long as it appears random to the attacker.

Game Hopping

- More complex proofs can be constructed using “game hopping”.
- If we can't give a reduction immediately, then we change the game slightly until we can give a reduction.
- For each change we have to show that the attacker's success probability doesn't increase significantly by the change.



Royal Holloway
University of London

Game Hopping

- What kind of changes can we make?
 - We can change the way in which certain elements are computed (with the same dist)
 - We can change the distribution of elements (if dist's are computationally indistinguishable)
 - We can change the “win condition” to exclude unlikely events
 - We can change the “win condition” to insist on an unlikely (but significant) event.



Royal Holloway
University of London

More Sophisticated Techniques

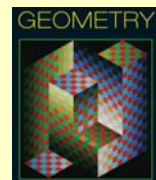
- Rewinding the attacker. The attacker is run twice with the same random tape and slightly different inputs.
 - Forking lemma for signatures and some zero-knowledge protocols.
- Non-black-box techniques (including self-awareness techniques)
 - Zero-knowledge protocols.



Royal Holloway
University of London

Arguments about proofs

- Taking another look



Royal Holloway
University of London

Taking another look

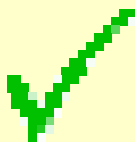
- In 2004, Koblitz and Menezes released the first in a sequence of papers entitled ‘Another Look at “Provable Security”’.



Taking another look



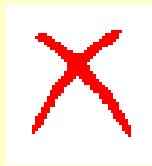
- “There are two unfortunate connotations of “proof”... the first is the notion of 100% security”.
- Security proofs are reductions.
- Breaking the ‘hard’ problem may or may not break the security of the scheme.



Taking another look



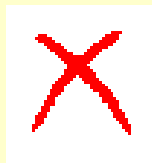
- “[T]he result is conditional in a strong sense ... the condition is of the sort ‘assuming that no one finds an improved algorithm for a certain math problem’”.
- The theorem means itself.
- Improved algorithms affect security regardless of proof.
- Security not a binary state.



Taking another look



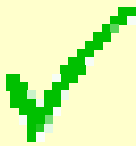
- “[A] provable security theorem only applies to attacks of a specified sort and says nothing about clever attacks that might not be included in the theorem”
- Security applies in the model.
- Model may be incorrectly interpreted or have inherent weaknesses.



Taking another look



- “If the only way to obtain a proof of security ... is to concoct an interactive math problem that people have no desire to study, then it seems to us that the [proof] is not very convincing”.
- Unfortunately, on the rise.



Taking another look



- “[Provable security experts] rarely read other authors’ papers carefully. As a result even the best authors sometimes publish papers with serious errors that go undetected for years”.

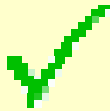


- “Do mistakes happen? Occasionally, though rarely.”

Taking another look



- “[Provable security experts] rarely read other authors’ papers carefully. As a result even the best authors sometimes publish papers with serious errors that go undetected for years”.
- In some conferences, it is rare that a correct proof is presented.



Playing tight

- How do we determine parameter sizes?
- Concrete security
- Do proofs have meaning?



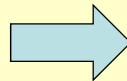
Tightness

- Security in a system typically depends on the size of the keys used – longer keys give more secure systems.
- If we consider polynomial-time attackers with asymptotically small success probabilities, then how do we determine correct parameter sizes?



Royal Holloway
University of London

Tightness



Size: L
Time: t
Success: ϵ

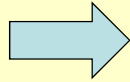
Size: L
Time: t
Success: ϵ

- If the hard problem is broken, then the scheme is broken.



Royal Holloway
University of London

Tightness



Size: L
 Time: t
 Success: ϵ

Size: $q_S L$
 Time: $t + O(\epsilon^{-2} \log \epsilon^{-1})$
 Success: $\epsilon^2/q_S - q_H/p$

- If the hard problem is broken, then the security of the scheme may be in doubt.



Royal Holloway
 University of London

Tightness



- [KM] argue that if we take the “tightness” of the security reduction into account when we choose key sizes, then the keys almost always become infeasible large.
- Alternatively, if we examine the proof at normal key sizes, then the proofs provide no security guarantees.

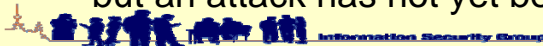


Royal Holloway
 University of London



Tightness

- [KM] provide seven interpretations of the meaning of a weak reduction:
 - #1 Even a non-tight reduction is better than nothing at all. One should ... derive some assurance from what one has.
 - #6 Perhaps the protocol is secure in practice, even though a tight reduction may simply not exist.
 - #7 Perhaps the protocol is in fact insecure, but an attack has not yet been discovered.



Royal Holloway
University of London

Tightness

- So how do we choose key sizes for a particular scheme?
- Ask an expert in the field!
- The looseness of a reduction does not mean that known techniques are more/less likely to break the scheme.
- Parameter sizes a quirk of history?



Royal Holloway
University of London

A tale of two signature schemes

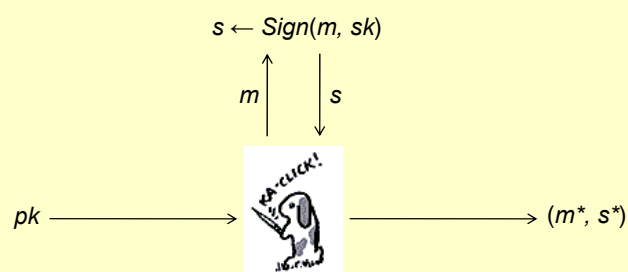
- Tightness and security in real schemes



Royal Holloway
University of London

The Magic Bit

- Attacker wins if s^* is a valid signature for m^* and the attacker never asked for the signature of m^* .



Royal Holloway
University of London

The Magic Bit

- One classic signature scheme was introduced by Rivest, Shamir and Adleman.



Royal Holloway
University of London

The Magic Bit

- Let p, q be large primes and let $n = pq$.
- Let e be co-prime to $(p-1)(q-1)$.
- Let d be such that $ed = 1 \pmod{(p-1)(q-1)}$.
- Then $m^{ed} = m \pmod{n}$.
- ... but we don't know how to compute m given only n , e , and $m^e \pmod{n}$.



Royal Holloway
University of London

The Magic Bit

- Generate $pk = (n, e)$ and $sk = (n, d)$.
- To sign a message m :
 1. Compute $y = \text{hash}(m)$.
 2. Compute $s = y^d \bmod n$.
- To verify a signature s on a message m :
 1. Compute $y = s^e \bmod n$.
 2. Check $y = \text{hash}(m)$.



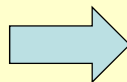
Royal Holloway
University of London

The Magic Bit

- A result of Coron states that if the hash function acts like a random function:



Size: L
Time: t
Success: ε



Size: L
Time: $\approx t$
Success: $\approx q_S \varepsilon$

where q_S is the number of signatures obtained.



Royal Holloway
University of London

The Magic Bit

- Katz and Wang proposed a new scheme:



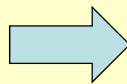
- To sign a message m :
 1. Pick a random bit b .
 2. Compute $y = \text{hash}(b, m)$.
 3. Compute $s = y^d \text{ mod } n$.
 4. Output (b, s) .



Royal Holloway
University of London

The Magic Bit

- Katz-Wang argued that if the hash function acts like a random function:



Size: L
Time: t
Success: ϵ

Size: L
Time: $\approx t$
Success: $\approx \epsilon$



Royal Holloway
University of London

The Magic Bit



- “It seems to us implausible that the Katz-Wang system could be ... more secure than the original RSA signature”.
- The schemes are very different.
- Probabilistic vs. Deterministic.
- Two signatures per message with no control over which one the oracle returns.



Royal Holloway
University of London

Conclusions

- Have we learnt anything through proofs?



Royal Holloway
University of London

Conclusions

- Koblitz and Menezes argue that provable security is no longer helping the field of practical cryptography.
- I would argue that it is helping the field develop at an accelerated rate.
- The criticisms of provable security seem to fall into two categories:



Royal Holloway
University of London

Conclusions

Interpretive

- Proof is a poor choice of words for a security reduction.
- Tightness of bounds is unhelpful in determining parameter sizes.
- Apparently similar schemes have unexpectedly differing security results.

Societal

- Results are rarely checked and are often rushed for conferences.
- Schemes without security results are rarely considered by most cryptographers.



Royal Holloway
University of London

Conclusions

- Researchers are basing cryptosystems on increasingly esoteric problems which will never be studied unless their security becomes critical to some business.
- Always good to challenge the *status quo* but I believe that the emperor is clothed on this occasion.



Royal Holloway
University of London

Conclusions



- “Instead of first setting up a natural cryptographic system and then trying to prove something about it, it has become increasingly common for protocol designers to start with a reductionist security objective that they want to achieve and use it to guide them in their construction of the scheme”.



Royal Holloway
University of London