On the Design and Implementation of Efficient Zero-Knowledge Proofs of Knowledge

SPEED-CC, Berlin (Germany), October 13th, 2009

Endre Bangerter\textsuperscript{1}, Stephan Krenn\textsuperscript{1,2}, Ahmad-Reza Sadeghi\textsuperscript{3}, Thomas Schneider\textsuperscript{3}, and Joe-Kai Tsay\textsuperscript{4}

\textsuperscript{1} Bern University of Applied Sciences (Switzerland)
\textsuperscript{2} University of Fribourg (Switzerland)
\textsuperscript{3} Ruhr-University Bochum (Germany)
\textsuperscript{4} Ecole Normale Supérieure de Cachan (France)
Why to Avoid ZK-PoK in Hidden Order Groups

SPEED-CC, Berlin (Germany), October 13\textsuperscript{th}, 2009

Endre Bangerter\textsuperscript{1}, Stephan Krenn\textsuperscript{1,2}, Ahmad-Reza Sadeghi\textsuperscript{3}, Thomas Schneider\textsuperscript{3}, and Joe-Kai Tsay\textsuperscript{4}

\textsuperscript{1} Bern University of Applied Sciences (Switzerland)  
\textsuperscript{2} University of Fribourg (Switzerland)  
\textsuperscript{3} Ruhr-University Bochum (Germany)  
\textsuperscript{4} Ecole Normale Supérieure de Cachan (France)
Outline

Proofs of knowledge in hidden order groups

Exact efficiency and security analysis

Conclusion
Introduction

**Proof of Knowledge**: Prover cannot cheat

**Zero-Knowledge**: Verifier cannot learn secret
Applications

Remote Authentication
(e.g. DAA)

Credential Systems
(e.g. idemix)
The Schnorr Protocol

I know \( x = \log_q y \).

\[
\begin{align*}
  r &\in_R \mathbb{Z} \\
  t &:= g^r \\
  s &:= r + cx
\end{align*}
\]

\[
\begin{align*}
  c &\in_R \mathbb{C} \\
  g^s &\overset{?}{=} ty^c
\end{align*}
\]
The Schnorr Protocol

\[ r \in_R \mathbb{Z} \]
\[ t := g^r \]
\[ s := r + cx \]

**I know** \( x = \log_g y \).

\[ c \in_R \mathbb{C} \]
\[ g^s = ty^c \]

**BUT:** We must use \( \mathbb{C} = \{0,1\} \)!
A Computationally Hard Problem

Given safe RSA modulus $n$, and $x, y \in \mathbb{Z}_n^*$, cannot compute $a, b, c, w$ such that $w^c = x^a y^b$ and $(c \nmid a \text{ or } c \nmid b)$.

holds under: Strong RSA Assumption

Given safe RSA modulus $n$, and $y \in \mathbb{Z}_n^*$, cannot compute $a, e \neq 1$ such that $a^e = y$. 
A Damgård/Fujisaki based Protocol

I know $x = \log_g y$.

\[ r, \bar{r}, \bar{x} \in_R \mathbb{Z} \]
\[ t := g^r \]
\[ \bar{y} := h_1^x \bar{h}^\bar{x} \]
\[ \bar{t} := h_1^r \bar{h}^\bar{r} \]
\[ s := r + cx \]
\[ \bar{s} := \bar{r} + c\bar{x} \]

With large challenge set.

\[ c \in_R C \]
\[ g^s \overset{?}{=} ty^c \]
\[ h_1^s \bar{h}^\bar{s} \overset{?}{=} \bar{t}y^c \]
Why it works...

\[ g^{s_i} = ty^{c_i} \quad i = 1,2 \]
\[ g^{\Delta s} = y^{\Delta c} \]
\[ x = \Delta s (\Delta c)^{-1} \]
Why it works...

\[ \begin{align*}
g^{s_i} &= ty^{c_i} \\
g^{\Delta s} &= y^{\Delta c} \\
x &= \Delta s \left( \Delta c \right)^{-1}
\end{align*} \]

\[ \begin{align*}
&\rightarrow \quad \bar{h}_1^{s_i} \bar{h}^{s_i} = \bar{t}y^{c_i} \quad i = 1,2 \\
&\rightarrow \quad \bar{h}_1^{\Delta s} \bar{h}^{\Delta s} = \bar{y}^{\Delta c} \quad \text{and} \quad \Delta c \mid \Delta s \\
&\rightarrow \quad x = \frac{\Delta s}{\Delta c}
\end{align*} \]
Outline

Proofs of knowledge in hidden order groups

Exact efficiency and security analysis

Conclusion
Intuitive Comparison

Schnorr protocol:
slow
looooong

DF-based protocol:
fast
elegant
A Closer Look

Common reference string

Only computationally sound

Bad complexity reductions
Bad Reductions

- Probability of breaking Strong RSA
- Probability of breaking the protocol
Is DAA broken?
Bad Reductions

Probability of breaking Strong RSA

Probability of breaking the protocol

loose reduction
Relative Costs

I know $x, r$, such that $y = g_1^{x^2} g_2^r$.

| $n_0$ | $|n| = 15528$ | $|n| = 2048$ | optimal | $|n|$ |
|-------|----------------|----------------|----------|-------|
| 1024  | 42.7           | 2.7            | 1.9      |       |
| 1280  | 24.0           | 1.7            | 1.1      |       |
| 1536  | 13.1           | 1.0            | 0.7      |       |
| 2048  | 5.6            | 0.6            | 0.3      |       |
So...
Sources of Inefficiency

- Complexity of proof goal
- Size of underlying group
- Flexibility of $|n|$
- Efficiency of math-library

Relative costs
Dependencies of Relative Costs

- Simplicity of proof goal
- Efficiency of math-library

- Decreasing size of underlying group
- Medium costs
- High costs
- Low costs
Outline

Proofs of knowledge in hidden order groups

Exact efficiency and security analysis

Conclusion
Conclusion

Crypto folklore

Design vs. implementation

RSA’s legacy

Schnorr

Damgård/Fujisaki

RSA 1
Conclusion

Crypto folklore

Design vs. implementation

RSA’s legacy

Schnorr

Damgård/Fujisaki

Java

RSA 1
Crypto folklore

Schnorr

Damgård/Fujisaki

Design vs. implementation

RSA’s legacy

1

Java
Conclusion

Crypto folklore
Schnorr
Damgård/Fujisaki

Design vs. implementation
Java

RSA's legacy
RSAPA
On the Design and Implementation of Efficient Zero-Knowledge Proofs of Knowledge

SPEED-C, Berlin (Germany), October 13th, 2009

Endre Bangerter¹, Stefan Krenn¹,², Ahmad-Reza Sadeghi³, Thomas Schneider³, and Joe-Kai Tsay⁴

¹ Bern University of Applied Sciences (Switzerland)
² University of Fribourg (Switzerland)
³ Ruhr-University Bochum (Germany)
⁴ Ecole Normale Supérieure de Cachan (France)