Motivation	Research Question	Results	Conclusion	Notes
00	0 000	0000	0	0

Construction of Universal Designated-Verifier Signatures and Identity-Based Signatures from Standard Signatures

Siamak Shahandashti¹ Rei Safavi-Naini²

¹SCSSE & CCISR, Uni Wollongong, Australia www.uow.edu.au/~sfs166

²Dept Comp Sci & iCIS, Uni Calgary, Canada www.cpsc.ucalgary.ca/~rei

PKC 2008

Universities of Wollongong and Calgary

UDVS & IBS from Signatures

Motivation	Research Question	Results	Conclusion	Notes
00	0000	0000	0	0

Outline

Motivation

Universal Designated-Verifier Signatures Identity-Based Signatures

Research Question

Research Question Formulation of Patterns

Results

Our UDVS Construction and Its Security

Our IBS Construction and Its Security

Conclusion

Concluding Remarks

Notes

Final Notes

Motivation	Research Question	Results	Conclusion	Notes
• O O	0 000	0000	0	0

What's a Universal Designated-Verifier Signature? a.k.a. UDVS

- Basically: a signature scheme with an extra functionality
- Goal: to protect user privacy when using credentials
- Idea: transform signature s.t. it only convinces a particular verifier



Universities of Wollongong and Calgary

э

不得下 イヨト イヨ

Motivation	Research Question	Results	Conclusion	Notes
• O O	0 000	0000	0	0

What's a Universal Designated-Verifier Signature? a.k.a. UDVS

- Basically: a signature scheme with an extra functionality
- Goal: to protect user privacy when using credentials
- Idea: transform signature s.t. it only convinces a particular verifier



UDVS & IBS from Signatures

Universities of Wollongong and Calgary

э

不得下 イヨト イヨ

Motivation	Research Question	Results	Conclusion	Notes
• O O	0 000	0000	0	0

What's a Universal Designated-Verifier Signature? a.k.a. UDVS

- Basically: a signature scheme with an extra functionality
- Goal: to protect user privacy when using credentials
- Idea: transform signature s.t. it only convinces a particular verifier



4 15 16 14 15

Motivation	Research Question	Results	Conclusion	Notes
0 0	0 000	0000	0	0

How can we construct a UDVS?

- $\hat{\sigma}$ is a designated-verifier non-interactive proof of holding a valid signature on *m*.
- Jakobsson et al's intuition to verifier designation: "Instead of proving X, Alice will prove the statement: Either X is true, or I am Bob."
- In the Random Oracle Model, non-interactive proofs can be constructed using Fiat-Shamir heuristic from ∑ protocols.
- So the only things we need are:
 - A ∑ protocol for proof of knowledge of a signature on a message, and
 - A ∑ protocol for proof of knowledge of the verifier's secret key.

э

A (10) A (10) A (10) A

○●	0	~
0 000 00		0

How can we construct a UDVS?

- $\hat{\sigma}$ is a designated-verifier non-interactive proof of holding a valid signature on *m*.
- Jakobsson et al's intuition to verifier designation: "Instead of proving X, Alice will prove the statement: Either X is true, or I am Bob."
- In the Random Oracle Model, non-interactive proofs can be constructed using Fiat-Shamir heuristic from ∑ protocols.
- So the only things we need are:
 - A Σ protocol for proof of knowledge of a signature on a message, and
 - A ∑ protocol for proof of knowledge of the verifier's secret key.

э

○●	0	~
0 000 00		0

How can we construct a UDVS?

- $\hat{\sigma}$ is a designated-verifier non-interactive proof of holding a valid signature on *m*.
- Jakobsson et al's intuition to verifier designation: "Instead of proving X, Alice will prove the statement: Either X is true, or I am Bob."
- In the Random Oracle Model, non-interactive proofs can be constructed using Fiat-Shamir heuristic from Σ protocols.
- So the only things we need are:
 - \blacktriangleright A Σ protocol for proof of knowledge of a signature on a message, and
 - A ∑ protocol for proof of knowledge of the verifier's secret key.

э

▲圖 ▶ ▲ 国 ▶ ▲ 国 ▶

Motivation	Research Question	Results	Conclusion	Notes
	0 000	0000	0	0

How can we construct a UDVS?

- $\hat{\sigma}$ is a designated-verifier non-interactive proof of holding a valid signature on *m*.
- Jakobsson et al's intuition to verifier designation: "Instead of proving X, Alice will prove the statement: Either X is true, or I am Bob."
- In the Random Oracle Model, non-interactive proofs can be constructed using Fiat-Shamir heuristic from Σ protocols.
- So the only things we need are:
 - A ∑ protocol for proof of knowledge of a signature on a message, and
 - A ∑ protocol for proof of knowledge of the verifier's secret key.

Motivation	Research Question	Results	Conclusion	Notes
○ ○ ●	0 000	0000	0	0

How can we construct an Identity-Based Signature? a.k.a. IBS



- σ is a signature on m that shows the signer has knowledge of usk
- In the Random Oracle Model, signatures can be constructed using Fiat-Shamir heuristic from ∑ protocols.
- So again the only thing we need is:
 - \blacktriangleright A Σ protocol for proof of knowledge of a signature on a message.

Universities of Wollongong and Calgary

э

Motivation	Research Question	Results	Conclusion	Notes
○ ○ ●	0 000	0000	0	0

How can we construct an Identity-Based Signature? a.k.a. IBS



- σ is a signature on m that shows the signer has knowledge of usk
- In the Random Oracle Model, signatures can be constructed using Fiat-Shamir heuristic from ∑ protocols.
- So again the only thing we need is:
 - \blacktriangleright A Σ protocol for proof of knowledge of a signature on a message.

Universities of Wollongong and Calgary

э

Motivation	Research Question	Results	Conclusion	Notes
○ ○ ●	0 000	0000	0	0

How can we construct an Identity-Based Signature? a.k.a. IBS



- σ is a signature on m that shows the signer has knowledge of usk
- In the Random Oracle Model, signatures can be constructed using Fiat-Shamir heuristic from ∑ protocols.
- So again the only thing we need is:
 - A ∑ protocol for proof of knowledge of a signature on a message.

э

Motivation	Research Question	Results	Conclusion	Notes
○ ○ ●	0 000	0000	0	0

How can we construct an Identity-Based Signature? a.k.a. IBS



- σ is a signature on m that shows the signer has knowledge of usk
- In the Random Oracle Model, signatures can be constructed using Fiat-Shamir heuristic from ∑ protocols.
- So again the only thing we need is:
 - A ∑ protocol for proof of knowledge of a signature on a message.

Universities of Wollongong and Calgary

4 B K 4 B K

00 0 000 0 0 0 000 000 00	Motivation	Research Question	Results	Conclusion	Notes
	00 0	000	0000	0	0

Research Question

So, What's the problem Then?

Although any NP relation has a Σ protocol, these generic protocols are normally not efficient!

Is there any more efficient way to do it?

UDVS & IBS from Signatures

Universities of Wollongong and Calgary

Motivation	Research Question	Results	Conclusion	Notes
00	○ ●OO	0000	0	0

Yes, There Is a Way!

We don't actually need strict honest-verifier zero-knowledge!

Example

Schnorr signature:

$$pk = (p, q, g, h = g^x), \quad \sigma = (c, z): \quad c = H(g^z \cdot h^{-c}, m)$$

To prove knowledge of a signature

- give out $aux = g^z \cdot h^{-c}$
- ▶ prove knowledge of $z: g^z = aux \cdot h^{H(aux,m)}$

Universities of Wollongong and Calgary

-

UDVS & IBS from Signatures

Motivation 00 0	Research Question ○ ●○○	Results	Conclusion o	Notes o
Formulation of Patterns				

Yes, There Is a Way!

We don't actually need strict honest-verifier zero-knowledge!

Example

Schnorr signature:

$$pk = (p, q, g, h = g^{x}), \quad \sigma = (c, z): \quad c = H(g^{z} \cdot h^{-c}, m)$$

To prove knowledge of a signature

- give out $aux = g^z \cdot h^{-c}$
- ▶ prove knowledge of $z: g^z = aux \cdot h^{H(aux,m)}$

Universities of Wollongong and Calgary

-

ヘロト 人間 とくほ とくほ とう

Motivation 00 0	Research Question ○ ●○○	Results	Conclusion o	Notes o
Formulation of Patterns				

Yes, There Is a Way!

We don't actually need strict honest-verifier zero-knowledge!

Example

Schnorr signature:

$$pk = (p, q, g, h = g^x), \quad \sigma = (c, z): \quad c = H(g^z \cdot h^{-c}, m)$$

To prove knowledge of a signature

• give out
$$aux = g^z \cdot h^{-c}$$

• prove knowledge of $z: g^z = aux \cdot h^{H(aux,m)}$

Universities of Wollongong and Calgary

Motivation	Research Question	Results	Conclusion	Notes
00	0 0●0	0000	0	0

Defining Class \mathbb{C} of Signatures

There exist Convert and Retrieve s.t.

$\tilde{\sigma} \gets \texttt{Convert}\left(pk, m, \sigma\right) \quad \Rightarrow \quad \sigma \gets \texttt{Retrieve}\left(pk, m, \tilde{\sigma}\right)$

and if $\tilde{\sigma} = (aux, pre)$ then there exists:

- ▶ An AuxSim that AuxSim (*pk*, *m*) simulates *aux*, and
- A Σ protocol for proof of knowledge of a *pre* for known *pk*, *m*, and *aux*.

(4月) トイラト イラト

Motivation	Research Question	Results	Conclusion	Notes
00	0 000	0000	0	0

Defining Class \mathbb{C} of Signatures

There exist Convert and Retrieve s.t.

 $\tilde{\sigma} \leftarrow \text{Convert}(pk, m, \sigma) \implies \sigma \leftarrow \text{Retrieve}(pk, m, \tilde{\sigma})$ and if $\tilde{\sigma} = (aux, pre)$ then there exists:

- ▶ An AuxSim that AuxSim (*pk*, *m*) simulates *aux*, and
- A ∑ protocol for proof of knowledge of a *pre* for known *pk*, *m*, and *aux*.

Universities of Wollongong and Calgary

- 4 周 ト 4 ヨ ト 4 ヨ ト

00 0 0000 0 0 0 0●0 00 0 0 0	Motivation	Research Question	Results	Conclusion	Notes
	00	0 0●0	0000	0	0

Defining Class \mathbb{C} of Signatures

There exist Convert and Retrieve s.t.

 $\tilde{\sigma} \leftarrow \text{Convert}(pk, m, \sigma) \implies \sigma \leftarrow \text{Retrieve}(pk, m, \tilde{\sigma})$ and if $\tilde{\sigma} = (aux, pre)$ then there exists:

- ▶ An AuxSim that AuxSim (*pk*, *m*) simulates *aux*, and
- A Σ protocol for proof of knowledge of a *pre* for known *pk*, *m*, and *aux*.

Universities of Wollongong and Calgary

00 0 0000 0 0 0 0●0 00 0 0 0	Motivation	Research Question	Results	Conclusion	Notes
	00	0 0●0	0000	0	0

Defining Class \mathbb{C} of Signatures

There exist Convert and Retrieve s.t.

 $\tilde{\sigma} \leftarrow \text{Convert}(pk, m, \sigma) \implies \sigma \leftarrow \text{Retrieve}(pk, m, \tilde{\sigma})$ and if $\tilde{\sigma} = (aux, pre)$ then there exists:

- ▶ An AuxSim that AuxSim (*pk*, *m*) simulates *aux*, and
- A Σ protocol for proof of knowledge of a *pre* for known *pk*, *m*, and *aux*.

Motivation	Research Question	Results	Conclusion	Notes
00	0 00•	0000	0	0

Which Signatures Does Class C Cover?

RSA-FDH, Schnorr, Modified ElGamal, Boneh-Lynn-Shacham, Boneh-Boyen, Cramer-Shoup, Camenisch-Lysyanskaya-02, Camenisch-Lysyanskaya-04, Goldwasser-Micali-Rivest, Gennaro-Halevi-Rabin, and Cramer-Shoup.

But not PSS of Bellare and Rogaway!

Universities of Wollongong and Calgary

UDVS & IBS from Signatures

Motivation	Research Question	Results	Conclusion	Notes
00	0 00•	0000	0	0

Which Signatures Does Class C Cover?

RSA-FDH, Schnorr, Modified ElGamal, Boneh-Lynn-Shacham, Boneh-Boyen, Cramer-Shoup, Camenisch-Lysyanskaya-02, Camenisch-Lysyanskaya-04, Goldwasser-Micali-Rivest, Gennaro-Halevi-Rabin, and Cramer-Shoup.

But not PSS of Bellare and Rogaway!

UDVS & IBS from Signatures

Universities of Wollongong and Calgary

Motivation	Research Question	Results	Conclusion	Notes
00	0 000	•000 00	0	0

How to Construct a UDVS from a Signature?

Use signature to sign

To designate: $(aux, pre) \leftarrow \text{Convert}(pk_s, m, \sigma)$ $\delta \leftarrow \text{SoK} \{(pre \lor sk_v) : \text{Valid}(pk_s, m, (aux, pre)), \text{Pair}(pk_v, sk_v)\}$ $\hat{\sigma} \leftarrow (aux, \delta)$

Verification is straightforward.

Universities of Wollongong and Calgary

(4月) キョン・チョン

Motivation	Research Question	Results	Conclusion	Notes
00 0	0 000	•000 00	0	0

How to Construct a UDVS from a Signature?

Use signature to sign

To designate:

 $\begin{aligned} (aux, pre) &\leftarrow \texttt{Convert}(pk_{\texttt{s}}, m, \sigma) \\ \delta &\leftarrow \texttt{SoK}\left\{(pre \lor sk_{\texttt{v}}) : \texttt{Valid}(pk_{\texttt{s}}, m, (aux, pre)), \texttt{Pair}(pk_{\texttt{v}}, sk_{\texttt{v}})\right\} \\ \hat{\sigma} &\leftarrow (aux, \delta) \end{aligned}$

Verification is straightforward.

Universities of Wollongong and Calgary

Motivation	Research Question	Results	Conclusion	Notes
00 0	0 000	•000 00	0	0

How to Construct a UDVS from a Signature?

Use signature to sign

To designate:

 $\begin{aligned} (aux, pre) &\leftarrow \texttt{Convert}\left(pk_{\texttt{s}}, m, \sigma\right) \\ \delta &\leftarrow \texttt{SoK}\left\{\left(pre \lor sk_{\texttt{v}}\right): \texttt{Valid}\left(pk_{\texttt{s}}, m, (aux, pre)\right), \texttt{Pair}\left(pk_{\texttt{v}}, sk_{\texttt{v}}\right)\right\} \\ \hat{\sigma} &\leftarrow (aux, \delta) \end{aligned}$

Verification is straightforward.

Universities of Wollongong and Calgary

00 0 000 0 0	Motivation	Research Question	Results	Conclusion	Notes
0 000 00	00	0 000	0000	0	0

Security of Our UDVS Construction

Let SS be any signature in \mathbb{C} and P_{SS} be its underlying problem. Also, let *KT* be any key type in \mathbb{K} and P_{KT} be its underlying problem. Then our UDVS construction:

- ▶ is *DV-unforgeable* if P_{SS} and P_{KT} are both hard.
- achieves non-transferability privacy.
- ▶ is *non-delegatable* if the challenge space of the proof protocol is big enough.

Universities of Wollongong and Calgary

A (10) A (10) A (10) A

Notivation Resea	rch Question Resu	Ilts Conclus	ion Notes
00 0 0 000	000	• •	0

Security of Our UDVS Construction

Let SS be any signature in \mathbb{C} and P_{SS} be its underlying problem. Also, let *KT* be any key type in \mathbb{K} and P_{KT} be its underlying problem. Then our UDVS construction:

- is *DV-unforgeable* if P_{SS} and P_{KT} are both hard.
- achieves non-transferability privacy.
- ▶ is *non-delegatable* if the challenge space of the proof protocol is big enough.

Universities of Wollongong and Calgary

A (1) A (2) A (2) A

Notivation Resea	rch Question Resu	Ilts Conclus	ion Notes
00 0 0 000	000	• •	0

Security of Our UDVS Construction

Let SS be any signature in \mathbb{C} and P_{SS} be its underlying problem. Also, let *KT* be any key type in \mathbb{K} and P_{KT} be its underlying problem. Then our UDVS construction:

- is *DV-unforgeable* if P_{SS} and P_{KT} are both hard.
- achieves non-transferability privacy.
- is non-delegatable if the challenge space of the proof protocol is big enough.

A (1) A (2) A (2) A

Notivation Resea	rch Question Resu	Ilts Conclus	ion Notes
00 0 0 000	000	• •	0

Security of Our UDVS Construction

Let SS be any signature in \mathbb{C} and P_{SS} be its underlying problem. Also, let *KT* be any key type in \mathbb{K} and P_{KT} be its underlying problem. Then our UDVS construction:

- is *DV-unforgeable* if P_{SS} and P_{KT} are both hard.
- achieves non-transferability privacy.
- is non-delegatable if the challenge space of the proof protocol is big enough.

Motivation 00	Research Question o ooo	Results ○○●○ ○○	Conclusion O	Notes o

How Good is Our Construction?

Comparison between Steinfeld et al's and our constructions

Schomo		Desi	g cost	â cizo	
Scheme	Tialu piobi.	off-line	on-line	0 5120	ND
DVSBM	BDH	none	1 pair.	1.0 kb	X
BLS+DL	CDH	2 pair.	1 mult.	5.3 kb	1
SchUDVS ₁	SDH	1 exp.	1 exp.	2.0 kb	X
$SchUDVS_2$	DL	2 exp.	1 exp.	1.5 kb	?
Schnorr+DL	DL	4 exp.	1 mult.	5.3 kb	1
RSAUDVS	RSA	1 exp.	2 exp.	11.6 kb	?
RSA-FDH+DL	RSA & DL	2 exp.	1 mult.	4.3 kb	1

ND: non-delegatability

Motivation oo	Research Question o ooo	Results ooo● ○○	Conclusion o	Notes o

Further Constructions

- universal multi-designated-verifier signatures: through non-interactive proof of knowledge of one out of n + 1 values: a (converted) signature and the secret keys of the n verifiers.
- designate more than one signature at once: e.g. to show at least k out of n certificates to a verifier, construct a non-interactive proof of knowledge of k + 1 out of n + 1 values: n (converted) signatures and the secret key of the verifier.
- a combination of the above two

(3)

Motivation	Research Question o ooo	Results ○○● ○○	Conclusion o	Notes o

Further Constructions

- universal multi-designated-verifier signatures: through non-interactive proof of knowledge of one out of n + 1 values: a (converted) signature and the secret keys of the n verifiers.
- designate more than one signature at once: e.g. to show at least k out of n certificates to a verifier, construct a non-interactive proof of knowledge of k + 1 out of n + 1 values: n (converted) signatures and the secret key of the verifier.
- a combination of the above two

Motivation	Research Question o ooo	Results ○○● ○○	Conclusion o	Notes o

Further Constructions

- universal multi-designated-verifier signatures: through non-interactive proof of knowledge of one out of n + 1 values: a (converted) signature and the secret keys of the n verifiers.
- designate more than one signature at once: e.g. to show at least k out of n certificates to a verifier, construct a non-interactive proof of knowledge of k + 1 out of n + 1 values: n (converted) signatures and the secret key of the verifier.
- a combination of the above two

Motivation	Research Question	Results	Conclusion	Notes
00	0000	0000 •0	0	0

How to Construct an IBS?

Use signature to issue user secret keys (signatures) on identities (messages) $usk \leftarrow SS.Sign(msk, id)$

```
To sign:

(aux, pre) \leftarrow \text{Convert}(mpk, id, usk)

\delta \leftarrow \text{SoK}\{pre : \text{Valid}(mpk, id, (aux, pre))\}(m)

\sigma \leftarrow (aux, \delta)
```

Verification is straightforward.

Universities of Wollongong and Calgary

Motivation	Research Question	Results	Conclusion	Notes
00	0000	0000 •0	0	0

How to Construct an IBS?

Use signature to issue user secret keys (signatures) on identities (messages) $usk \leftarrow SS.Sign(msk, id)$

```
To sign:

(aux, pre) \leftarrow \texttt{Convert}(mpk, id, usk)

\delta \leftarrow \texttt{SoK}\{pre : \texttt{Valid}(mpk, id, (aux, pre))\}(m)

\sigma \leftarrow (aux, \delta)
```

Verification is straightforward.

Universities of Wollongong and Calgary

Motivation oo	Research Question	Results ○○○○ ●○	Conclusion o	Notes o

How to Construct an IBS?

Use signature to issue user secret keys (signatures) on identities (messages) $usk \leftarrow SS.Sign(msk, id)$

```
To sign:

(aux, pre) \leftarrow \texttt{Convert}(mpk, id, usk)

\delta \leftarrow \texttt{SoK}\{pre : \texttt{Valid}(mpk, id, (aux, pre))\}(m)

\sigma \leftarrow (aux, \delta)
```

Verification is straightforward.

Motivation	Research Question	Results	Conclusion	Notes
00	0 000	0000	0	0

Security and Further Construction

Let SS be a standard signature in \mathbb{C} and P_{SS} be its underlying problem. Our IBS construction is ID-EUF-CMA-secure if P_{SS} is hard.

Further constructions:

- hierarchical identity-based signatures
- identity-based universal designated verifier signatures
- identity-based ring signatures

A 3 1 A 3 1

Motivation	Research Question	Results	Conclusion	Notes
00	0 000	0000 0 0	0	0

Security and Further Construction

Let SS be a standard signature in \mathbb{C} and P_{SS} be its underlying problem. Our IBS construction is ID-EUF-CMA-secure if P_{SS} is hard.

Further constructions:

- hierarchical identity-based signatures
- identity-based universal designated verifier signatures
- identity-based ring signatures

Motivation	Research Question	Results	Conclusion	Notes
00	0 000	0000	•	0

Summary

Our constructions:

- are almost generic, yet comparable in size and cost.
- are provably non-delegatable and also offer signer-verifier setting independence.
- can be extended to generic UMDVS, HIBS, IBUDVS, and IBRS.

However:

- our security proofs are in the Random Oracle Model.
- our security reductions are not tight.

A 3 1 A 3 1

Motivation	Research Question	Results	Conclusion	Notes
00	0 000	0000	•	0

Summary

Our constructions:

- are almost generic, yet comparable in size and cost.
- are provably non-delegatable and also offer signer-verifier setting independence.
- can be extended to generic UMDVS, HIBS, IBUDVS, and IBRS.

However:

- our security proofs are in the Random Oracle Model.
- our security reductions are not tight.

(3)

Motivation	Research Question	Results	Conclusion	Notes
00	0000	0000	•	0

Summary

Our constructions:

- are almost generic, yet comparable in size and cost.
- are provably non-delegatable and also offer signer-verifier setting independence.
- can be extended to generic UMDVS, HIBS, IBUDVS, and IBRS.

However:

- our security proofs are in the Random Oracle Model.
- our security reductions are not tight.

4 B 6 4 B 6

Motivation	Research Question	Results	Conclusion	Notes
00	0000	0000	•	0

Summary

Our constructions:

- are almost generic, yet comparable in size and cost.
- are provably non-delegatable and also offer signer-verifier setting independence.
- can be extended to generic UMDVS, HIBS, IBUDVS, and IBRS.

However:

- our security proofs are in the Random Oracle Model.
- our security reductions are not tight.

Universities of Wollongong and Calgary

Motivation	Research Question	Results	Conclusion	Notes
00	0000	0000	•	0

Summary

Our constructions:

- are almost generic, yet comparable in size and cost.
- are provably non-delegatable and also offer signer-verifier setting independence.
- can be extended to generic UMDVS, HIBS, IBUDVS, and IBRS.

However:

- our security proofs are in the Random Oracle Model.
- our security reductions are not tight.

Universities of Wollongong and Calgary

Motivation	Research Question	Results	Conclusion	Notes
00	0 000	0000	0	٠

Final Notes

Acknowledgment and Further Reading

Thanks to:

- ► *i*CORE Information Security Lab of Uni of Calgary
- Shaoquan Jiang and anonymous reviewers of PKC '08

Full paper:

Shahandashti and Safavi-Naini. Construction of Universal Designated-Verifier Signatures and Identity-Based Signatures from Standard Signatures. Cryptology ePrint Archive, Report 2007/462 (2007). http://eprint.iacr.org/2007/462

・回り ・ヨト ・ヨト

Motivation	Research Question	Results	Conclusion	Notes
00	0 000	0000	0	•

Final Notes

Acknowledgment and Further Reading

Thanks to:

- ► *i*CORE Information Security Lab of Uni of Calgary
- Shaoquan Jiang and anonymous reviewers of PKC '08

Full paper:

Shahandashti and Safavi-Naini.

Construction of Universal Designated-Verifier Signatures and Identity-Based Signatures from Standard Signatures. *Cryptology ePrint Archive*, Report 2007/462 (2007). http://eprint.iacr.org/2007/462

.