IMPROVED STRONGLY DENIABLE AUTHENTICATED KEY EXCHANGES FOR SECURE MESSAGING

Nik Unger
and
Ian Goldberg
Secure Messaging
Secure Messaging

- Anonymous Authentication (e.g., OTR, Signal)
- Deniable Authentication
- "All-Verifier" Authentication

Confidentiality:
- Plaintext

End-to-End Zone:
- TLS to Server
- Authentication
Cryptographic signatures demonstrate that Democratic National Committee Chairman Donna Brazile is wrong when she suggests the WikiLeaks emails were altered and that she did not send an email tipping off Democratic presidential nominee Hillary Clinton to debate questions.

Many email systems use a verification system called DomainKeys Identified Mail (DKIM) that shows whether an email has been changed. It uses a key stored on the email server that sent the email, so it can’t be forged.

HillaryClinton.com uses Gmail to handle its mail and uses DKIM. Staffer Jennifer Palmieri, using her HillaryClinton.com email, replied to a Brazile email warning that Brazile was “worried” about Clinton’s ability to answer a question about the
Deniable Messaging

A

Crypto Magic

B

<B> there’s a protest about it tomorrow
<B> want to go?
<A> Yes!
<B> ok, no phones

C
Deniable Messaging

There's a protest about it tomorrow. Do you want to go?

Yes!

OK, no phones.
Deniable Messaging...?
Offline vs. Online Deniability

**Offline Deniability**
- A
- B
- Crypto Magic

- <B> there’s a protest about it tomorrow
- <B> want to go?
- <A> Yes!
- <B> ok, no phones

**Online Deniability**
Deniable Messaging...?

• See Appendix A
  − Attacks on OTRv3 and Signal

• Also see ia.cr/2018/424:

On The Use of Remote Attestation to Break and Repair Deniability

Lachlan J. Gunn
Aalto University
lachlan.gunn@aalto.fi

Ricardo Vieitez Parra
Aalto University
ricardo.vieitezparra@aalto.fi

N. Asokan
Aalto University
asokan@acm.org
Deniable Messaging
Deniable Messaging
In This Paper

• Two new efficient key exchange protocols

Interactive

Non-interactive
Security Properties

- Confidentiality
- Mutual authentication
- Forward secrecy
- Contributiveness
- Offline and online deniability
Crypto Toolbox

- Identity key (long-term asymmetric)
- Ephemeral key (short-term asymmetric)
- Shared session key (symmetric)
- Diffie-Hellman shared secret
**Crypto Toolbox**

- **Signature**
  - Create: need private key
  - Verify: need public key

- **MAC**
  - Create: need one key
  - Verify: need all public keys

- **Ring signature**
  - Create: need one private key
  - Verify: need all public keys
Crypto Toolbox

- ID key
- Eph. key
- Sym. key
- Diffie-Hellman shared secret
- Signature
- MAC
- Ring signature
Deniable Authenticated Key Exchanges

Secure messaging protocol
DAKEZ

Shared key (\(\cdot\)):
DAKEZ: Authentication

- ID key
- Eph. key
- Sym. key
- Diffie-Hellman shared secret
- Signature
- MAC
- Ring signature

Nobody else knows or , so they know

Shared key ( ):
DAKEZ: Authentication

Shared key (\(\text{\{}\text{\}}\)):

Nobody else knows or, so they know.
DAKEZ: Offline Deniability

- ID key
- Eph. key
- Sym. key
- Diffie-Hellman shared secret
- Signature
- MAC
- Ring signature
- Shared key ( ):
DAKEZ: Online Deniability

Shared key ( ):
Mobile?
Mobile Use

“A”

“Prekeys”

Recipient ID

B

Message

Message
ZDH

Shared key (\(\bullet\)): 

- ID key
- Eph. key
- Sym. key
- Diffie-Hellman shared secret
- Signature
- MAC
- Ring signature
ZDH: Authentication

- ID key
- Eph. key
- Sym. key
- Diffie-Hellman shared secret
- Signature
- MAC
- Ring signature

Nobody else knows so any reader must know

Shared key ( ):

&
Weak Forward Secrecy
(Like Signal, originally)

(Time passes)

Collect
XZDH
Is This Secure?
"Yes."
1. INTRODUCTION

Cryptography is commonly used to secure private communications over the Internet. One way is to try to protect casual personal conversations, in a way that mimics the idea of a conversation it, as no cryptographic evidence of it can exist. However, the current secure messaging tools only allow for limited deniability, where the privacy and security of the participants engaging in a conversation can be compromised. With this in mind, we designed OTRv4 as an alternative tool to enhance the privacy of conversations.
## Performance

<table>
<thead>
<tr>
<th></th>
<th>SIGMA-R (OTRv3)</th>
<th>DAKEZ (OTRv4)</th>
<th>3DH</th>
<th>ZDH</th>
<th>X3DH (Signal)</th>
<th>XZDH (OTRv4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key Gen. (ms)</td>
<td>0.0240</td>
<td>0.0440</td>
<td>0.0228</td>
<td>0.0429</td>
<td>0.0240</td>
<td>0.0444</td>
</tr>
<tr>
<td>Key Exch. (ms)</td>
<td>0.3478</td>
<td>1.094</td>
<td>0.4229</td>
<td>0.778</td>
<td>0.5533</td>
<td>0.9217</td>
</tr>
<tr>
<td>ID Key (bytes)</td>
<td>32</td>
<td>32</td>
<td>32</td>
<td>32</td>
<td>32</td>
<td>32</td>
</tr>
<tr>
<td>Prekey (bytes)</td>
<td>-</td>
<td>-</td>
<td>32</td>
<td>32</td>
<td>32 &amp; 96</td>
<td>32 &amp; 96</td>
</tr>
<tr>
<td>Key Exch. (bytes)</td>
<td>272</td>
<td>464</td>
<td>80</td>
<td>304</td>
<td>80</td>
<td>304</td>
</tr>
</tbody>
</table>
Extras in the Paper

But Wait...
There's MORE!
Extras in the Paper

Quantum-resistant transitional security

Efficient dual-receiver encryption

Defeating key-compromise impersonation

Implementation details & advice
Summary

- New key exchanges: DAKEZ, (X)ZDH
- Secure connection, eponymous, no all-verifier authentication required? Use these!
- Code & data: crysp.org/software/dakez_xzdh
- Come see OTRv4 at HotPETs
- Coming soon: group messaging

Thank you!
njunger@uwaterloo.ca
You’ve Activated My Bonus Slides!!!
Limited Online Deniability

"Prekeys"

Recipient ID

Auth with

, Auth, Msg

A

B
RSDAKE and Spawn

- Standard model $\rightarrow$ Random oracle model
  - Obscure assumptions $\rightarrow$ common assumptions
  - Seconds $\rightarrow$ milliseconds
  - Improved security (contributiveness, forward secrecy)
- RSDAKE $\rightarrow$ DAKEZ
- Spawn $\rightarrow$ ZDH
### Table 1. Comparison of DAKE features, computational performance, and size requirements

<table>
<thead>
<tr>
<th></th>
<th>ECDH</th>
<th>3DH</th>
<th>X3DH</th>
<th>SIGMA-R</th>
<th>$\Phi_{idre}$</th>
<th>RSDAKE</th>
<th>Spawn</th>
<th>DAKEZ</th>
<th>Spawn$^+$</th>
<th>ZDH</th>
<th>XZDH</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Offline Deniable</strong></td>
<td>●</td>
<td>●</td>
<td>○</td>
<td>○</td>
<td>●</td>
<td>●</td>
<td>●</td>
<td>●</td>
<td>●</td>
<td>●</td>
<td>●</td>
</tr>
<tr>
<td><strong>Online Deniable</strong></td>
<td>●</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>●</td>
<td>●</td>
<td>○</td>
<td>●</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td><strong>Authenticated</strong></td>
<td>-</td>
<td>●</td>
<td>●</td>
<td>●</td>
<td>●</td>
<td>●</td>
<td>●</td>
<td>●</td>
<td>●</td>
<td>●</td>
<td>●</td>
</tr>
<tr>
<td><strong>Non-Interactive</strong></td>
<td>●</td>
<td>●</td>
<td>●</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>●</td>
<td>●</td>
<td>●</td>
</tr>
<tr>
<td><strong>Forward Secrecy</strong></td>
<td>-</td>
<td>-</td>
<td>○</td>
<td>●</td>
<td>●</td>
<td>●</td>
<td>-</td>
<td>●</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td><strong>Proof Model</strong></td>
<td>SM</td>
<td>ROM</td>
<td>ROM</td>
<td>ROM</td>
<td>SM</td>
<td>SM</td>
<td>SM</td>
<td>ROM</td>
<td>ROM</td>
<td>ROM</td>
<td>ROM</td>
</tr>
<tr>
<td><strong>Public Key</strong></td>
<td>-</td>
<td>0.0228</td>
<td>0.0240</td>
<td>0.0240</td>
<td>0.40</td>
<td>206</td>
<td>206</td>
<td>0.0440</td>
<td>0.0429</td>
<td>0.0441</td>
<td>0.0444</td>
</tr>
<tr>
<td>Generation [ms]</td>
<td></td>
<td>(0.0012)</td>
<td>(0.0013)</td>
<td>(0.0012)</td>
<td>(0.01)</td>
<td>(8)</td>
<td>(4)</td>
<td>(0.0016)</td>
<td>(0.0016)</td>
<td>(0.0018)</td>
<td>(0.0017)</td>
</tr>
<tr>
<td><strong>Exchange [ms]</strong></td>
<td>0.1733</td>
<td>0.4229</td>
<td>0.5533</td>
<td>0.3478</td>
<td>13</td>
<td>6630</td>
<td>3390</td>
<td>1.094</td>
<td>1.3683</td>
<td>0.778</td>
<td>0.9217</td>
</tr>
<tr>
<td></td>
<td>(0.0033)</td>
<td>(0.0050)</td>
<td>(0.0056)</td>
<td>(0.0048)</td>
<td>(2)</td>
<td>(50)</td>
<td>(20)</td>
<td>(0.014)</td>
<td>(0.0082)</td>
<td>(0.013)</td>
<td>(0.0069)</td>
</tr>
<tr>
<td><strong>Flows</strong></td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>9</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td><strong>Public Key [B]</strong></td>
<td>-</td>
<td>32</td>
<td>32</td>
<td>32</td>
<td>415</td>
<td>395</td>
<td>992</td>
<td>32</td>
<td>32</td>
<td>32</td>
<td>32</td>
</tr>
<tr>
<td><strong>Prekey [B]</strong></td>
<td>-</td>
<td>32</td>
<td>32+96</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>938</td>
<td>-</td>
<td>32</td>
<td>32</td>
<td>32+96</td>
</tr>
<tr>
<td><strong>Exchange [B]</strong></td>
<td>64</td>
<td>80</td>
<td>80</td>
<td>272</td>
<td>5140</td>
<td>7598</td>
<td>73763</td>
<td>464</td>
<td>512</td>
<td>304</td>
<td>304</td>
</tr>
</tbody>
</table>

- = provides property; ○ = partially provides property; - = does not provide property / not applicable; SM = standard model; ROM = random oracle model. Standard deviations are in parentheses. “Forward secrecy” is the strong variant [14] (all schemes have weak forward secrecy). Prekeys are listed as (one-time)+(signed) sizes.
Signal Deniability

3DH

3DH

X3DH

X3DH

IK_A

IK_B

IK_A

IK_B

EK_A

EK_B

EK_A

EK_B

SPK_B

OTK_B
Lack of Contributiveness

• **Problems** with non-contributory:
  - Can coerce a client to use a known secret
  - Can use a secret known to a third-party, allowing them to decrypt without their consent

• **Non-problems** with non-contributory:
  - Contributiveness does not prevent desirable bits
  - Contributiveness does not defend against weak PRNGs
ZDH

Shared key (      ):

ID key
Eph. key
Sym. key
Diffie-Hellman shared secret
Signature
MAC
Ring signature
ZDH: Authentication

- ID key
- Eph. key
- Sym. key
- Diffie-Hellman shared secret
- Signature
- MAC
- Ring signature

Shared key (\(\cdots\)):

Nobody else knows or , so they know . They also know .
Mitigating KCI Attacks

Shared key (\(\mathcal{K}\)):
Online Deniability Attack for Signal

- (Alice is coerced by Judson)
- Alice downloads Bob’s prekey: $IK_B$, $SPK_B$, $\text{Sig}(IK_B, \text{Encode}(SPK_B))$
- Judson generates key pair with public $EK_A$
- Alice provably reveals $\text{DH}(IK_A, SPK_A)$
- Alice sends $EK_A$ to Bob
- Judson can compute the secret, Alice cannot
Quantum Transitional Security

- Authenticate quantum KEM, like CECPK1

<table>
<thead>
<tr>
<th>Scheme</th>
<th>$\Delta$ Time [ms]</th>
<th>$PQ_I$ [bytes]</th>
<th>$QR$ [bytes]</th>
</tr>
</thead>
<tbody>
<tr>
<td>New Hope</td>
<td>+0.0542 (0.0041)</td>
<td>+1824</td>
<td>+2048</td>
</tr>
<tr>
<td>SIDH</td>
<td>+63.8 (1.5)</td>
<td>+768</td>
<td>+768</td>
</tr>
</tbody>
</table>
Fig. 2. The DAKEZ protocol. $\Phi$ is shared session state. The shared secret is $k = \text{KDF}(g^{ir} \parallel Q_k)$. 

I

Choose $i \leftarrow \mathbb{Z}_q$

$(PQ_I, SQ_I) \leftarrow \text{QRGen}_I()$

Verify proof from $\mathcal{R}$

Compute $Q_k = \text{QRKey}_I(SQ_I, Q_R)$

Compute $k$

RSign($g^R, R, \{g^I, g^R, g^i\}$, “0”“I”“R”“g^i”“g^i”“PQ_I”“Q_R”“$\Phi$”)$

\rightarrow$

Q_R$

R

Choose $r \leftarrow \mathbb{Z}_q$

$(Q_R, Q_k) \leftarrow \text{QRGen}_R(PQ_I)$

Compute $k$

Erase $r$ and $Q_k$

RSign($g^I, I, \{g^I, g^R, g^r\}$, “1”“I”“R”“g^i”“g^r”“PQ_I”“Q_R”“$\Phi$”)$

\rightarrow$

Verify proof from I

$\Phi$ is shared session state. The shared secret is $k = \text{KDF}(g^{ir} \parallel Q_k)$. 

47
Fig. 4. A ZDH/XZDH exchange. $\Phi$ is shared session state. $\kappa = \text{KDF}_1(g^i_r \parallel g^{\Gamma_r} \parallel g^{I_r} \parallel Q_k)$, $M_k = \text{KDF}_2(\kappa)$ and the shared secret is $k = \text{KDF}_3(\kappa)$. Shaded terms are used in XZDH only, and omitted for ZDH. In XZDH, $g^{\Gamma}$ is a reusable signed prekey.