# Pseudonymous Software Development and Strong Distribution

V. Alex Brennen
codepoet@dublin.ie

# PSDASD

- What is Pseudonymous Development?
  - Tied to a cyberspace identity
  - Not tied to a meatspace identity
- What is Strong Distribution?
  - A distribution model which is cryptographically strong
  - Software and Communications protected through the use of strong public key cryptography
  - Most commonly PGP or x509

# PSDASD

- Why do we need It?
  - Patent System has long been broken
  - Developers may now face imprisonment
    - DMCA (Reverse Engineering)
    - DRM
    - SkipJack
  - Developers may now face lawsuits
    - Trade Secrets
    - DMCA
      - RIAA
      - MPAA
    - Unintended Usage

# PSDASD

- Other Sources of Chilling Effects
  - Employer Mission Confusion
    - Public University or Private Incubator?
    - Researcher or Entrepreneur?
    - 9-5 Employee or Slave Property?
  - American Entitlement Liability
    - Grand Theft Auto
    - Columbine High School

# PSDASD

- Two adversaries
  - The US Gov't
    - The US Gov't is evil, ever present, and ever monitoring
      - You can't win against them
      - There are no secrets from them
  - Corporations
    - Limited by their own property
    - Limited by borders
    - Attempt to get the Gov't to act for them, but it's difficult

# PSDASD

- Choose Just Not To Contribute
  - Don't write any code
  - Send small patches and ask not to be named
- Try and Contribute in Secret
  - Separate your on-line identity
  - Contribute under a pseudonym
  - But what if you're discovered and exposed?
  - The subject of this talk!

# PSDASD


http://www.freestateproject.org/

# PSDASD

- Core Components Necessary
  - Key Server Infrastructure
    - A few different networks
  - Anonymous Email Infrastructure
    - Mixmaster anonymous remailers
  - Anonymous Posting Infrastructure
    - email to usenet gateways
  - Onion Routing Downloading Infrastructure
    - Tor

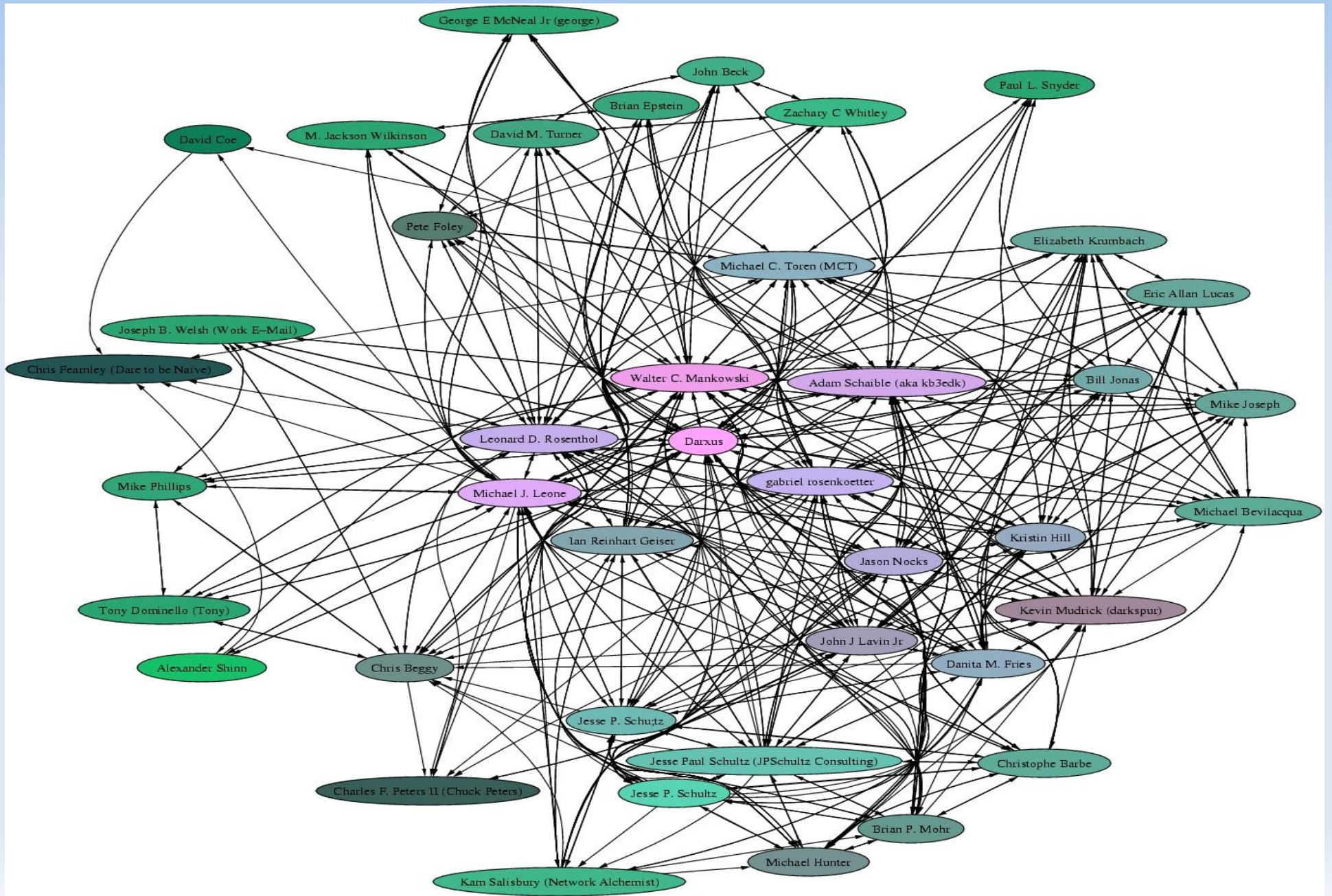
# PSDASD

- Create an Identity
  - openPGP key with no contact information
  - Integrate it with a Web of Trust

```
pub   2048R/939C69E9 2006-07-23 [expires: 2007-07-23]
uid                  Pseudonymous Hacker
sub   2048R/A5BC4FE5 2006-07-23 [expires: 2007-07-23]
```

```
pub  1024D/DE885DD3 2002-04-10 Sander Striker <striker@apache.org>
sig         E2226795 2002-05-01   Justin R. Erenkrantz
sig 3       DE885DD3 2002-04-10   Sander Striker
sig         CD4DF205 2002-05-28   Wolfram Schlich
sig         E005C9CB 2002-11-17   Greg Stein
sig         CC8B0F7E 2002-11-18   Aaron Bannert
sig         DFEAC4B9 2002-11-19   David N. Welton
sig 2       82AB7BD1 2002-11-17   Cliff Woolley
sig 2       13046155 2002-11-28   Thom May
sig 3       19311B00 2002-11-17   Chuck Murcko
sig 3       F894BE12 2002-11-17   Brian William Fitzpatrick
sig 3       5C1C3AD7 2002-11-18   David Reid
sig 3       E04F9A89 2002-11-18   Roy T. Fielding
sig 3       CC78C893 2002-11-19   Rich Bowen
sig 3       08C975E5 2002-11-21   Jim Jagielski
sig 3       F88341D9 2002-11-18   Lars Eilebrecht
sig 3       187BD68D 2002-11-21   Ben Hyde
sig 3       49A563D9 2002-11-23   Mark Cox
...more signatures redacted...
```

# PSDASD


George E McNeal Jr (george)
John Beck
Paul L. Snyder
Brian Epstein
Zachary C Whitley
David Coe
M. Jackson Wilkinson
David M. Turner
Pete Foley
Elizabeth Krumbach
Michael C. Toren (MCT)
Eric Allan Lucas
Joseph B. Welsh (Work E–Mail)
Chris Fearnley (Dare to be Naïve)
Walter C. Mankowski
Adam Schaible (aka kb3edk)
Bill Jonas
Mike Joseph
Leonard D. Rosenthol
Darxus
Mike Phillips
gabriel rosenkoetter
Michael J. Leone
Michael Bevilacqua
Ian Reinhart Geiser
Kristin Hill
Jason Nocks
Tony Dominello (Tony)
Kevin Mudrick (darkspur)
John J Lavin Jr
Alexander Shinn
Chris Beggy
Danita M. Fries
Jesse P. Schultz
Jesse Paul Schultz (JPSchultz Consulting)
Christophe Barbe
Charles F. Peters II (Chuck Peters)
Jesse P. Schultz
Brian P. Mohr
Michael Hunter
Kam Salisbury (Network Alchemist)

# PSDASD

- Step 1:  Write the Software
- Step 2:  Sign A Software Archive
- Step 3:  Post Your Public Key
- Step 4:  Distribute Software over Pseudonymous Network
- Step 5:  Publish a Method of Communicating With You
- Step 6:  Rinse and Repeat

# PPSDASD

- Step 1:  Write The Software
  - That's the hard part

# PSDF

- Step 2: Sign a Software Archive
  - gpg -a -o prog.tar.bz2.sig –detach-sig prog.tar.bz2
  - Provides Protection on the Server
  - Provides Protection in Transit
  - Provides Protection for the Customer

# PSDASD

- Step 3:  Post Your Public Key
  - Many Keyserver Networks
    - MIT Keyserver
    - subkeys
    - CryptNET
  - Single Web Site
  - p2p Network
  - Newsgroup

# PSDASD

- Step 4: Distribute Software over Pseudonymous Network
  - Tor
  - email to usenet
  - Mirrors to pick it up
  - Major distribution sites
    - SourceForge

# PSDASD

- Step 5: Publish a Method of Communicating With You
  - alt.anonymous
  - Steganography
  - Avoid Personal Email
  - Email Lists with Archives OK

# PSDASD

- Step 6:  Rinse and Repeat
  - New releases made the same way
  - Safely use method for years
  - Alternative method available if a system gets shut down
  - Sneaker net leaks

# PSDASD

- The patch life cycle
  - User gets software and public key
  - Writes patch
    - encrypts with public key
    - posts cyphertext it in public place
  - Developer Discovers Cyphertext
    - Decrypt
    - Processes patch
    - Signed security advistory
    - New release w/ fix

# PSDASD

- Egoboo
  - Developer asks Potential Employer to Encrypt Secret with project public key and provide cyphertext
  - Developer decrypts secret
  - Message Identifying Developer Signed with Project Keypair

# PSDASD

- Arbitrary Statements about Arbitrary Content
  - A hash representation of anything can be digitally signed
  - Signatures can be circulated in detached form
  - Pseudonymous Security Audits
  - Non-Pseudonymous Security Audits

# PSDASD

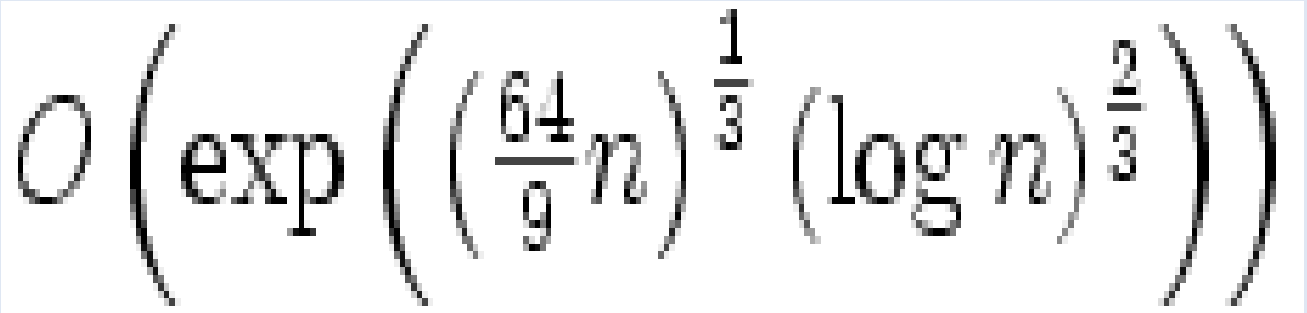
- Project Forking
  - Easy as Generating New Keypair
  - Signature on Keypair Lend Credibility
    - Old project keypair signature
    - Developer signature

# PSDASD

- Compromise
  - Identity
  - Project Keypair
  - Developer Keypair
  - Keys in Web of Trust
  - Keypair Revocation is Important
  - Possession of Keypair proof of involvement

# PSDASD

- Breaking a Keypair with Factoring
  - Government can do it



# PSDASD

- Ring Signatures
  - Someone in a Group
    - Signer Hard to Identifiable
  - Secret Leaking Protocols Can Be Instructive

# PSDASD

## The Strong Distribution HOWTO

http://cryptnet.net/fdp/crypto/strong_distro.html

## Guerrilla Software Development HOWTO

http://cryptnet.net/fdp/crypto/guerrilla-devl.html

## The Keysigning Party HOWTO

http://cryptnet.net/fdp/crypto/gpg-party.html

# PSDASD

V. Alex Brennen

codepoet@dublin.ie