Crypto horror stories

Daniel J. Bernstein

University of Illinois at Chicago
& Technische Universiteit Eindhoven

Horror story 1

RC4

RC4 stream cipher: The beginning

1987: Ron Rivest designs RC4. Does not publish it.

RC4 stream cipher: The beginning

1987: Ron Rivest designs RC4. Does not publish it.

1992: U.S. National Security Agency (NSA)
makes a deal with Software Publishers Association.

"NSA allows encryption ... The U.S. Department of State will grant export permission to any program that uses the RC2 or RC4 data-encryption algorithm with a key size of less than 40 bits."

RC4 stream cipher: The leak

1994: Someone anonymously posts RC4 source code.

New York Times: “Widespread dissemination could compromise the long-term effectiveness of the system . . . [RC4] has become the de facto coding standard for many popular software programs including Microsoft Windows, Apple’s Macintosh operating system and Lotus Notes. . . . ‘I have been told it was part of this deal that RC4 be kept confidential,’ Jim Bidzos, president of RSA, said.”

RC4 stream cipher: Used in SSL

1994: Netscape introduces SSL (“Secure Sockets Layer”) web browser and server “based on RSA Data Security technology”.

SSL supports many options. RC4 is fastest cipher in SSL.

RC4 stream cipher: Used in SSL, and broken

1994: Netscape introduces SSL (“Secure Sockets Layer”) web browser and server “based on RSA Data Security technology”.

SSL supports many options. RC4 is fastest cipher in SSL.

1995: Finney posts some examples of SSL ciphertexts. Back–Byers–Young, Doligez, Back–Brooks extract plaintexts.

RC4 stream cipher: Used in SSL, and broken

1994: Netscape introduces SSL (“Secure Sockets Layer”) web browser and server “based on RSA Data Security technology”.

SSL supports many options. RC4 is fastest cipher in SSL.

1995: Finney posts some examples of SSL ciphertexts. Back–Byers–Young, Doligez, Back–Brooks extract plaintexts.

Fix: RC4-128?

RC4 stream cipher: Used in SSL, and broken

1994: Netscape introduces SSL ("Secure Sockets Layer") web browser and server "based on RSA Data Security technology".

SSL supports many options. RC4 is fastest cipher in SSL.

1995: Finney posts some examples of SSL ciphertexts. Back–Byers–Young, Doligez, Back–Brooks extract plaintexts.

Fix: RC4-128? Unacceptable:
1995 Roos shows that RC4 fails a basic definition of cipher security.

RC4 stream cipher: The end?

So the crypto community throws away 40-bit keys?
And throws away RC4?

RC4 stream cipher: The end?

So the crypto community throws away 40-bit keys?
And throws away RC4?

Here's what actually happens.

RC4 stream cipher: The end?

So the crypto community throws away 40-bit keys?
And throws away RC4?

Here's what actually happens.

1997: IEEE standardizes WEP ("Wired Equivalent Privacy") for 802.11 wireless networks. WEP uses RC4 for encryption.

RC4 stream cipher: The end?

So the crypto community throws away 40-bit keys?
And throws away RC4?

Here's what actually happens.

1997: IEEE standardizes WEP ("Wired Equivalent Privacy") for 802.11 wireless networks. WEP uses RC4 for encryption.

1999: TLS ("Transport Layer Security"), new version of SSL. RC4 is fastest cipher in TLS. TLS still supports "export keys".

RC4 stream cipher: Great, we can write papers

More RC4 cryptanalysis: 1995 Wagner, 1997 Golic, 1998 Knudsen–Meier–Preneel–Rijmen–Verdoolaege, 2000 Golic, 2000 Fluhrer–McGrew, 2001 Mantin–Shamir, 2001 Fluhrer–Mantin–Shamir, 2001 Stubblefield–Ioannidis–Rubin.

Example of real-world damage:
RC4 key-output correlations ⇒ practical attacks on WEP.

RC4 stream cipher: Not dead yet!

2001 Rivest response: RC4 is safe in TLS.

“Applications which pre-process the encryption key and IV by using hashing and/or which discard the first 256 bytes of pseudo-random output should be considered secure from the proposed attacks. . . . The ‘heart’ of RC4 is its exceptionally simple and extremely efficient pseudo-random generator. . . . RC4 is likely to remain the algorithm of choice for many applications and embedded systems.”

RC4 stream cipher: More papers; more damage

2002 Hulton, 2002 Mironov, 2002 Pudovkina, 2003 Bittau, 2003 Pudovkina, 2004 Paul–Preneel, 2004 KoreK, 2004 Devine, 2005 Maximov, 2005 Mantin, 2005 d'Otreppe, 2006 Klein, 2006 Doroshenko–Ryabko, 2006 Chaabouni.

RC4 stream cipher: More papers; more damage

2002 Hulton, 2002 Mironov, 2002 Pudovkina, 2003 Bittau, 2003 Pudovkina, 2004 Paul–Preneel, 2004 KoreK, 2004 Devine, 2005 Maximov, 2005 Mantin, 2005 d'Otreppe, 2006 Klein, 2006 Doroshenko–Ryabko, 2006 Chaabouni.

WEP blamed for 2007 theft of 45 million credit-card numbers from T. J. Maxx. Subsequent lawsuit settled for $40900000.

RC4 stream cipher: Even more papers

2007 Paul–Maitra–Srivastava, 2007 Paul–Rathi–Maitra, 2007 Paul–Maitra, 2007 Vaudenay–Vuagnoux, 2007 Tews–Weinmann–Pyshkin, 2007 Tomasevic–Bojanic–Nieto-Taladriz, 2007 Maitra–Paul, 2008 Basu–Ganguly–Maitra–Paul, 2008 Biham–Carmeli, 2008 Golic–Morgari, 2008 Maximov–Khovratovich, 2008 Akgun–Kavak–Demirci, 2008 Maitra–Paul. 2008 Beck–Tews, 2009 Basu–Maitra–Paul–Talukdar, 2010 Sepehrdad–Vaudenay–Vuagnoux, 2010 Vuagnoux, 2011 Maitra–Paul–Sen Gupta, 2011 Sen Gupta–Maitra–Paul–Sarkar, 2011 Paul–Maitra book.

RC4 stream cipher: Resurgence in popularity

2012 Akamai blog entry: “Up to 75% of SSL-enabled web sites are vulnerable [to BEAST] . . . OpenSSL v0.9.8w is the current version in broad use and it only supports TLS v1.0. . . . the interim fix is to prefer the RC4-128 cipher for TLS v1.0 and SSL v3. . . . RC4-128 is faster and cheaper in processor time . . . approximately 15% of SSL/TLS negotiations on the Akamai platform use RC4 . . . most browsers can support the RC4 fix for BEAST.”

HOW TO KILL A
ZOMBIE
(BEFORE THEY EAT YOUR BRAINS)
SHOT GUN
SHOVEL
CHAINSAW
STEP 1: CHOOSE YOUR WEAPON
STEP 2: AIM FOR THE HEAD
STEP 3: WHATEVER YOU DO DON'T MISS

RC4 stream cipher: How to kill a zombie

2013 Lv–Zhang–Lin, 2013 Lv–Lin, 2013 Sen Gupta–Maitra–Meier–Paul–Sarkar, 2013 Sarkar–Sen Gupta–Paul–Maitra, 2013 Isobe–Ohigashi–Watanabe–Morii, 2013 AlFardan–Bernstein–Paterson–Poettering–Schuldt, 2014 Paterson–Strefler, 2015 Sepherdad–Sušil–Vaudenay–Vuagnoux, 2015 Mantin “Bar Mitzvah”, 2015 Garman–Paterson–van der Merwe “RC4 must die”, 2015 Vanhoef–Piessens “RC4 no more”.

RC4 stream cipher: How to kill a zombie

2013 Lv–Zhang–Lin, 2013 Lv–Lin, 2013 Sen Gupta–Maitra–Meier–Paul–Sarkar, 2013 Sarkar–Sen Gupta–Paul–Maitra, 2013 Isobe–Ohigashi–Watanabe–Morii, 2013 AlFardan–Bernstein–Paterson–Poettering–Schuldt, 2014 Paterson–Strefler, 2015 Sepherdad–Sušil–Vaudenay–Vuagnoux, 2015 Mantin "Bar Mitzvah", 2015 Garman–Paterson–van der Merwe "RC4 must die", 2015 Vanhoef–Piessens "RC4 no more".

IETF RFC 7465 ("RC4 die die die") prohibits RC4 in TLS.

RC4 stream cipher: How to kill a zombie

2013 Lv–Zhang–Lin, 2013 Lv–Lin, 2013 Sen Gupta–Maitra–Meier–Paul–Sarkar, 2013 Sarkar–Sen Gupta–Paul–Maitra, 2013 Isobe–Ohigashi–Watanabe–Morii, 2013 AlFardan–Bernstein–Paterson–Poettering–Schuldt, 2014 Paterson–Strefler, 2015 Sepherdad–Sušil–Vaudenay–Vuagnoux, 2015 Mantin "Bar Mitzvah", 2015 Garman–Paterson–van der Merwe "RC4 must die", 2015 Vanhoef–Piessens "RC4 no more".

IETF RFC 7465 ("RC4 die die die") prohibits RC4 in TLS.

2015.09: Google, Microsoft, Mozilla announce agreement to turn off RC4 in subsequent browser updates.

It's not just RC4

Some ongoing problems illustrated by this story:

- Incompetent risk management.

It's not just RC4

Some ongoing problems illustrated by this story:

- Incompetent risk management.
- Security being damaged by the pursuit of performance.

It's not just RC4

Some ongoing problems illustrated by this story:

- Incompetent risk management.
- Security being damaged by the pursuit of performance.
- Security being damaged by algorithm “agility”.

It’s not just RC4

Some ongoing problems illustrated by this story:

- Incompetent risk management.
- Security being damaged by the pursuit of performance.
- Security being damaged by algorithm “agility”.
- Security being damaged intentionally by NSA.

It's not just RC4

Some ongoing problems illustrated by this story:

- Incompetent risk management.
- Security being damaged by the pursuit of performance.
- Security being damaged by algorithm "agility".
- Security being damaged intentionally by NSA.
- Academic incentives negatively correlated with security.

It’s not just RC4

Some ongoing problems illustrated by this story:

- Incompetent risk management.
- Security being damaged by the pursuit of performance.
- Security being damaged by algorithm “agility”.
- Security being damaged intentionally by NSA.
- Academic incentives negatively correlated with security.
- Standardization incentives negatively correlated with security.

It’s not just RC4

Some ongoing problems illustrated by this story:

- Incompetent risk management.
- Security being damaged by the pursuit of performance.
- Security being damaged by algorithm “agility”.
- Security being damaged intentionally by NSA.
- Academic incentives negatively correlated with security.
- Standardization incentives negatively correlated with security.
- Industrial incentives negatively correlated with security.

It's not just RC4

Some ongoing problems illustrated by this story:

- Incompetent risk management.
- Security being damaged by the pursuit of performance.
- Security being damaged by algorithm "agility".
- Security being damaged intentionally by NSA.
- Academic incentives negatively correlated with security.
- Standardization incentives negatively correlated with security.
- Industrial incentives negatively correlated with security.

This year NSA is pushing new low-security ciphers through ISO.

Horror story 2

Timing attacks

Timing attacks: Early history

1970s: TENEX operating system compares
user-supplied string against secret password
one character at a time, stopping at first difference:

- `AAAAAA` vs. `SECRET`: stop at 1.
- `SAAAAA` vs. `SECRET`: stop at 2.
- `SEAAAA` vs. `SECRET`: stop at 3.

Attacker watches comparison time, deduces position of difference.
A few hundred tries reveal secret password.

Timing attacks: Example of some bad code

How typical software checks 16-byte authenticator:

```
for (i = 0;i < 16;++i)
  if (x[i] != y[i]) return 0;
return 1;
```

Fix, eliminating information flow to timings:

```
diff = 0;
for (i = 0;i < 16;++i)
  diff |= x[i] ^ y[i];
return (1 & ((diff - 1) >> 8)) - 1;
```

Timing attacks: Do they actually work?

Objection: “Timings are noisy!”

Timing attacks: Do they actually work?

Objection: “Timings are noisy!”

Answer #1: Does noise stop *all* attacks?
To guarantee security, defender must block *all* information flow.

Timing attacks: Do they actually work?

Objection: “Timings are noisy!”

Answer #1: Does noise stop *all* attacks?
To guarantee security, defender must block *all* information flow.

Answer #2: Attacker uses statistics to eliminate noise.

Timing attacks: Do they actually work?

Objection: “Timings are noisy!”

Answer #1: Does noise stop *all* attacks?
To guarantee security, defender must block *all* information flow.

Answer #2: Attacker uses statistics to eliminate noise.

Answer #3, what the 1970s attackers actually did:
Cross page boundary, inducing page faults, to amplify timing signal.

Timing attacks: Defenders don’t learn

1996 Kocher pointed out timing attacks on cryptographic key bits.

Timing attacks: Defenders don't learn

1996 Kocher pointed out timing attacks on cryptographic key bits.

2008 RFC 5246 "The Transport Layer Security (TLS) Protocol, Version 1.2": "This leaves a small timing channel, since MAC performance depends to some extent on the size of the data fragment, but it is not believed to be large enough to be exploitable, due to the large block size of existing MACs and the small size of the timing signal."

Timing attacks: Defenders don’t learn

1996 Kocher pointed out timing attacks on cryptographic key bits.

2008 RFC 5246 “The Transport Layer Security (TLS) Protocol, Version 1.2”: “This leaves a small timing channel, since MAC performance depends to some extent on the size of the data fragment, but it is not believed to be large enough to be exploitable, due to the large block size of existing MACs and the small size of the timing signal.”

2013 AlFardan–Paterson “Lucky Thirteen: breaking the TLS and DTLS record protocols”: exploit these timings; steal plaintext.

Timing attacks: Sophistication increases

2005 Tromer–Osvik–Shamir: 65ms to steal Linux AES key used for hard-disk encryption. Idea: *AES key influences CPU cache timings*. Attack process on same CPU but without privileges.

Timing attacks: Sophistication increases

2005 Tromer–Osvik–Shamir: 65ms to steal Linux AES key used for hard-disk encryption. Idea: *AES key influences CPU cache timings*. Attack process on same CPU but without privileges.

2014 Irazoqui–Inci–Eisenbarth–Sunar "Wait a minute! A fast, Cross-VM attack on AES" recovers "the AES keys of OpenSSL 1.0.1 running inside the victim VM" *despite VMware virtualization*.

Timing attacks: Sophistication increases

2005 Tromer–Osvik–Shamir: 65ms to steal Linux AES key used for hard-disk encryption. Idea: *AES key influences CPU cache timings*. Attack process on same CPU but without privileges.

2014 Irazoqui–Inci–Eisenbarth–Sunar “Wait a minute! A fast, Cross-VM attack on AES” recovers “the AES keys of OpenSSL 1.0.1 running inside the victim VM” *despite VMware virtualization*.

2016 García–Brumley–Yarom stole DSA host key from OpenSSH server via timings of OpenSSL.

Horror story 3

The attackers

What are the attackers doing?

2012.09: I gave a talk "Cryptography for the paranoid": "They're monitoring *everything* we do on the Internet. And they're *changing* packets and faking *web pages* in transit without our even noticing. And they have huge armies of *computers* analyzing everything."

What are the attackers doing?

2012.09: I gave a talk "Cryptography for the paranoid": "They're monitoring *everything* we do on the Internet. And they're *changing* packets and faking *web pages* in transit without our even noticing. And they have huge armies of *computers* analyzing everything."

What about encryption?

What are the attackers doing?

2012.09: I gave a talk "Cryptography for the paranoid": "They're monitoring *everything* we do on the Internet. And they're *changing* packets and faking *web pages* in transit without our even noticing. And they have huge armies of *computers* analyzing everything."

What about encryption?

"They're *recording* everything. Even if they don't understand it today, they'll keep looking at it for *years* until they understand it. They have huge armies of *mathematicians* analyzing it. And they're working on building *quantum computers*."

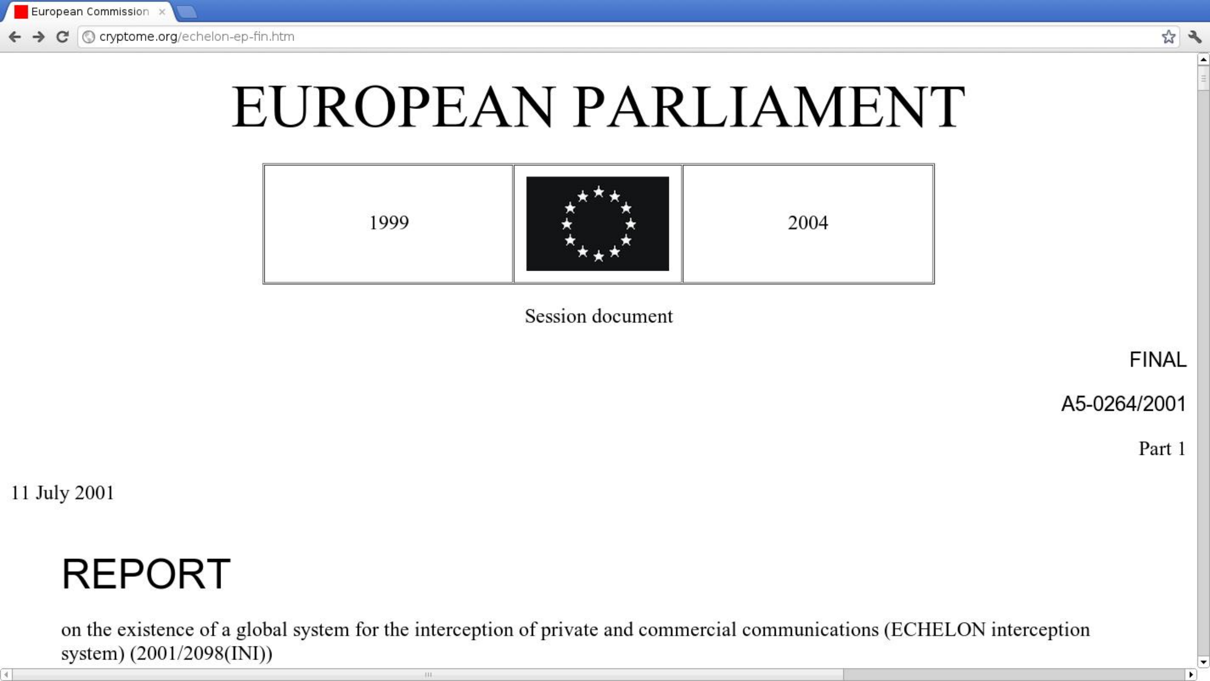
What are the attackers doing?

2012.09: I gave a talk "Cryptography for the paranoid": "They're monitoring *everything* we do on the Internet. And they're *changing* packets and faking *web pages* in transit without our even noticing. And they have huge armies of *computers* analyzing everything."

What about encryption?

"They're *recording* everything. Even if they don't understand it today, they'll keep looking at it for *years* until they understand it. They have huge armies of *mathematicians* analyzing it. And they're working on building *quantum computers*."

This was pre-Snowden. What was my evidence?



EUROPEAN PARLIAMENT

1999		2004

Session document

FINAL

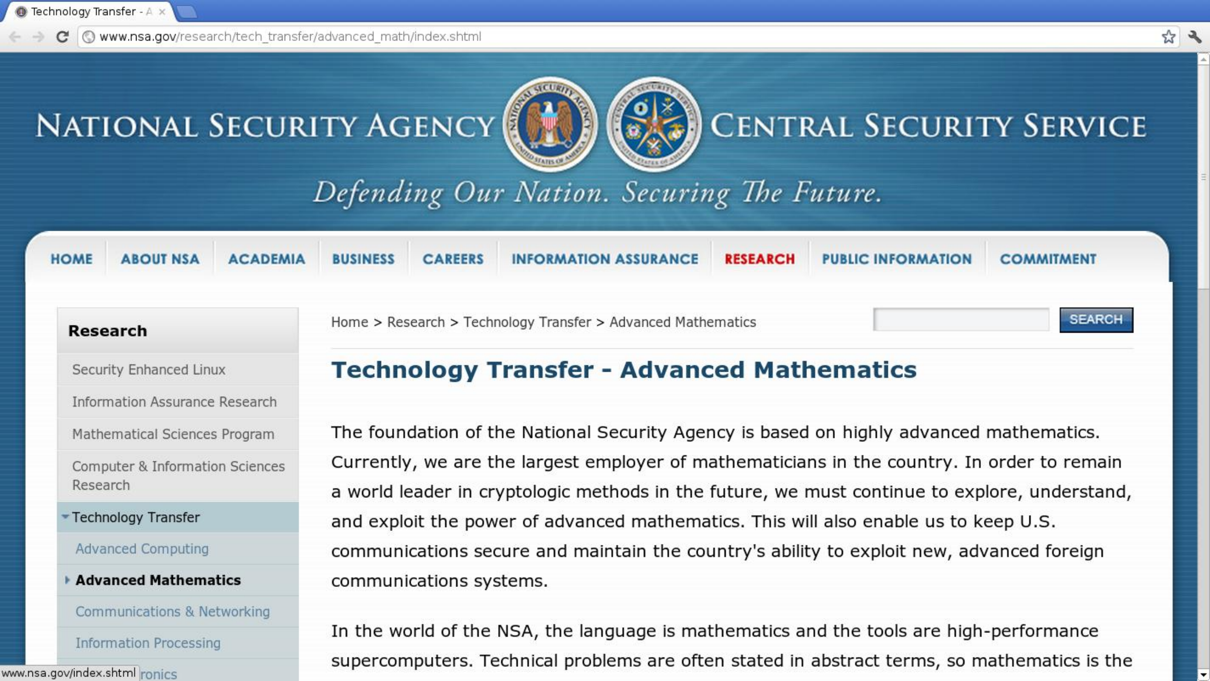
A5-0264/2001

Part 1

11 July 2001

REPORT

on the existence of a global system for the interception of private and commercial communications (ECHELON interception system) (2001/2098(INI))



Technology Transfer - A
www.nsa.gov/research/tech_transfer/advanced_math/index.shtml

NATIONAL SECURITY AGENCY CENTRAL SECURITY SERVICE

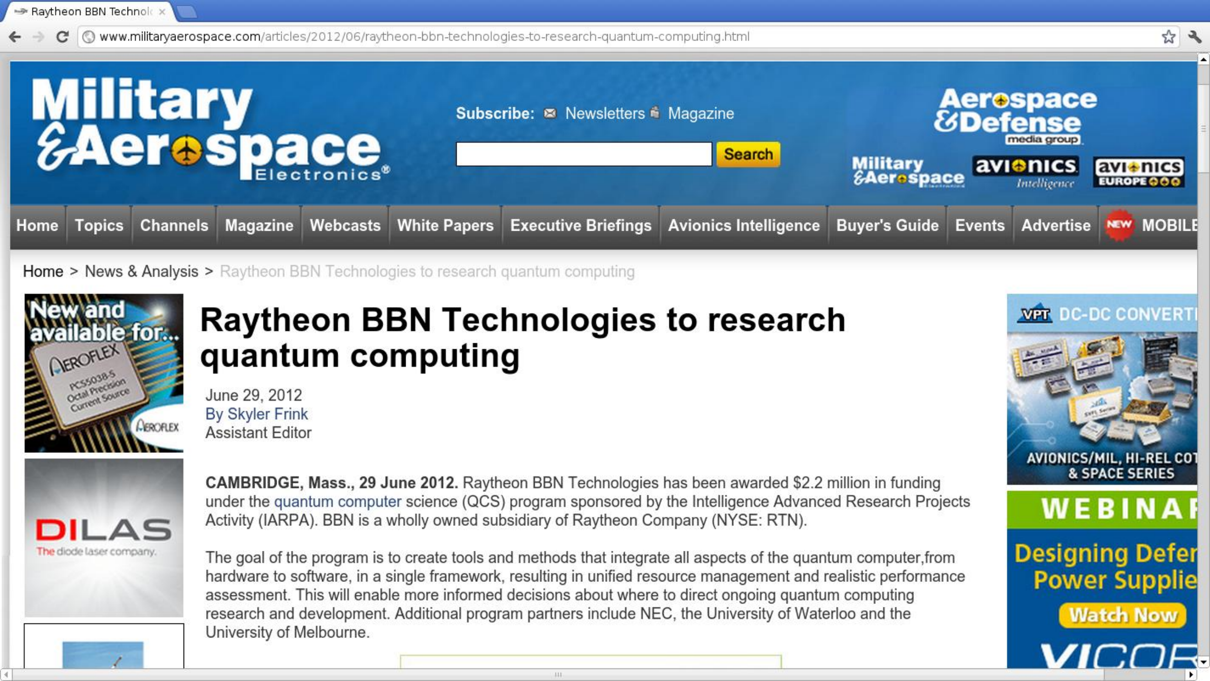
Defending Our Nation. Securing The Future.

Home > Research > Technology Transfer > Advanced Mathematics
SEARCH

Technology Transfer - Advanced Mathematics

The foundation of the National Security Agency is based on highly advanced mathematics. Currently, we are the largest employer of mathematicians in the country. In order to remain a world leader in cryptologic methods in the future, we must continue to explore, understand, and exploit the power of advanced mathematics. This will also enable us to keep U.S. communications secure and maintain the country's ability to exploit new, advanced foreign communications systems.

In the world of the NSA, the language is mathematics and the tools are high-performance supercomputers. Technical problems are often stated in abstract terms, so mathematics is the

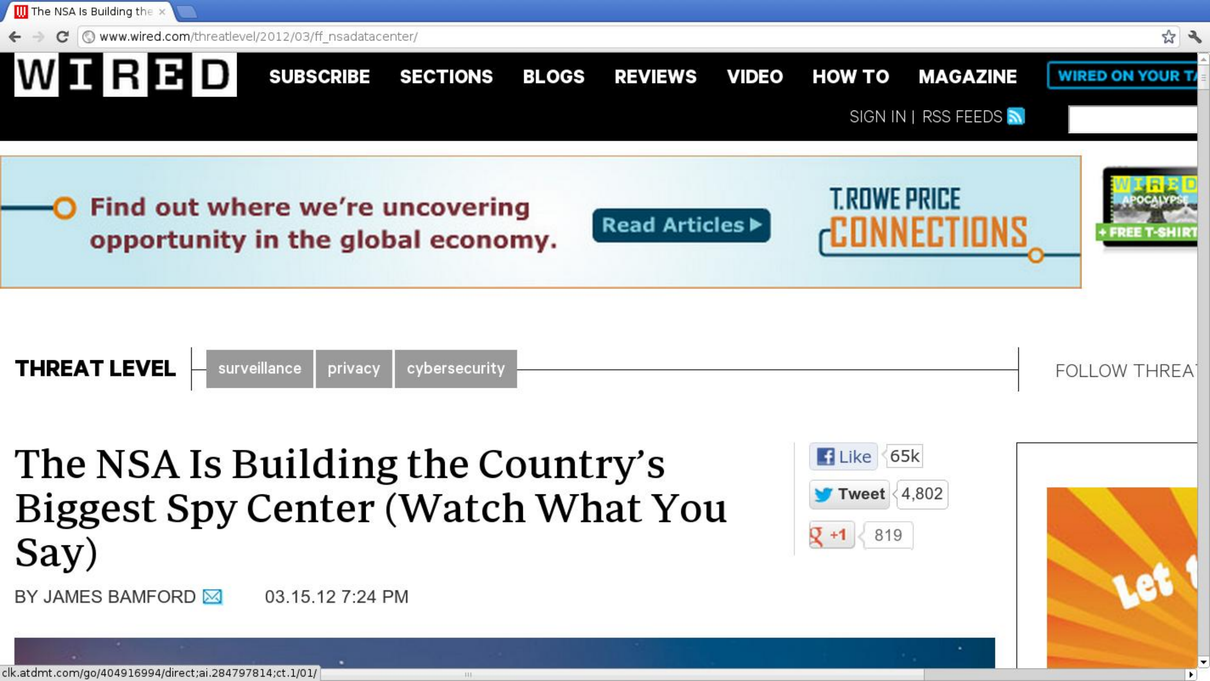

Raytheon BBN Technol
www.militaryaerospace.com/articles/2012/06/raytheon-bbn-technologies-to-research-quantum-computing.html
Military & Aerospace Electronics®
Subscribe: Newsletters Magazine
Search
Aerospace & Defense media group
Military & Aerospace
avionics Intelligence
avionics EUROPE
Home
Topics
Channels
Magazine
Webcasts
White Papers
Executive Briefings
Avionics Intelligence
Buyer's Guide
Events
Advertise
NEW
MOBILE
Home > News & Analysis > Raytheon BBN Technologies to research quantum computing

Raytheon BBN Technologies to research quantum computing

June 29, 2012
By Skyler Frink
Assistant Editor

CAMBRIDGE, Mass., 29 June 2012. Raytheon BBN Technologies has been awarded $2.2 million in funding under the quantum computer science (QCS) program sponsored by the Intelligence Advanced Research Projects Activity (IARPA). BBN is a wholly owned subsidiary of Raytheon Company (NYSE: RTN).

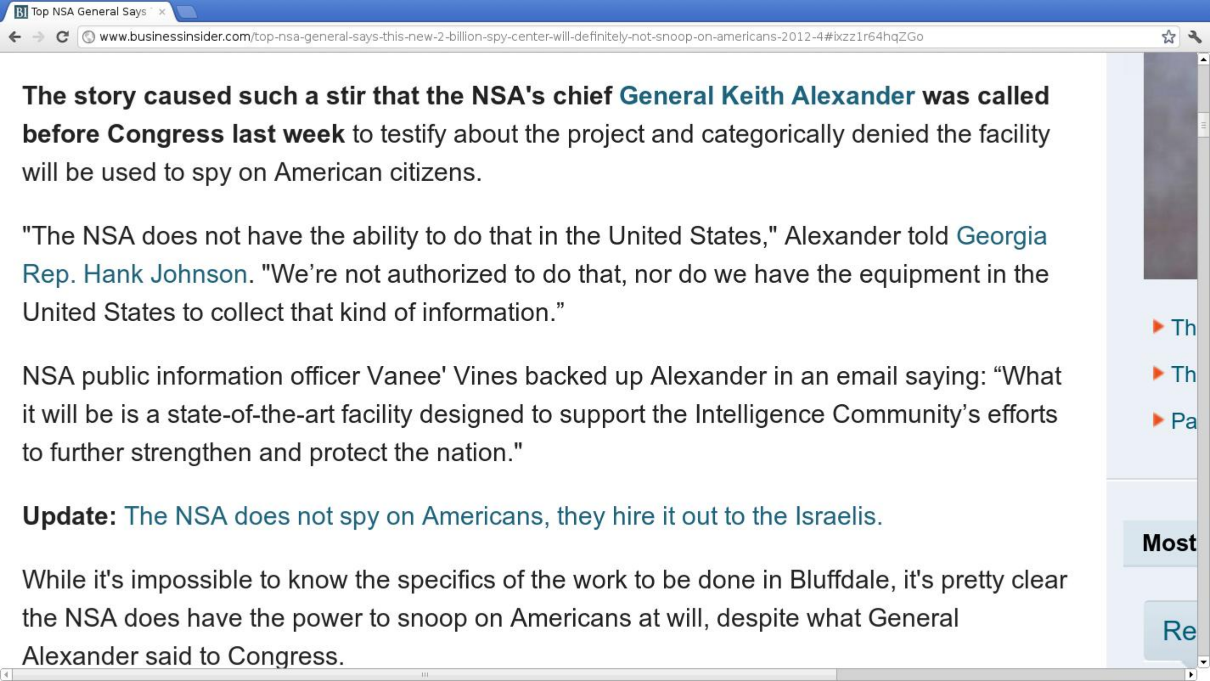
The goal of the program is to create tools and methods that integrate all aspects of the quantum computer,from hardware to software, in a single framework, resulting in unified resource management and realistic performance assessment. This will enable more informed decisions about where to direct ongoing quantum computing research and development. Additional program partners include NEC, the University of Waterloo and the University of Melbourne.



THREAT LEVEL

The NSA Is Building the Country's Biggest Spy Center (Watch What You Say)

BY JAMES BAMFORD ✉ 03.15.12 7:24 PM


Top NSA General Says
www.businessinsider.com/top-nsa-general-says-this-new-2-billion-spy-center-will-definitely-not-snoop-on-americans-2012-4#ixzz1r64hqZGo
Th
Th
Pa
Most
Re

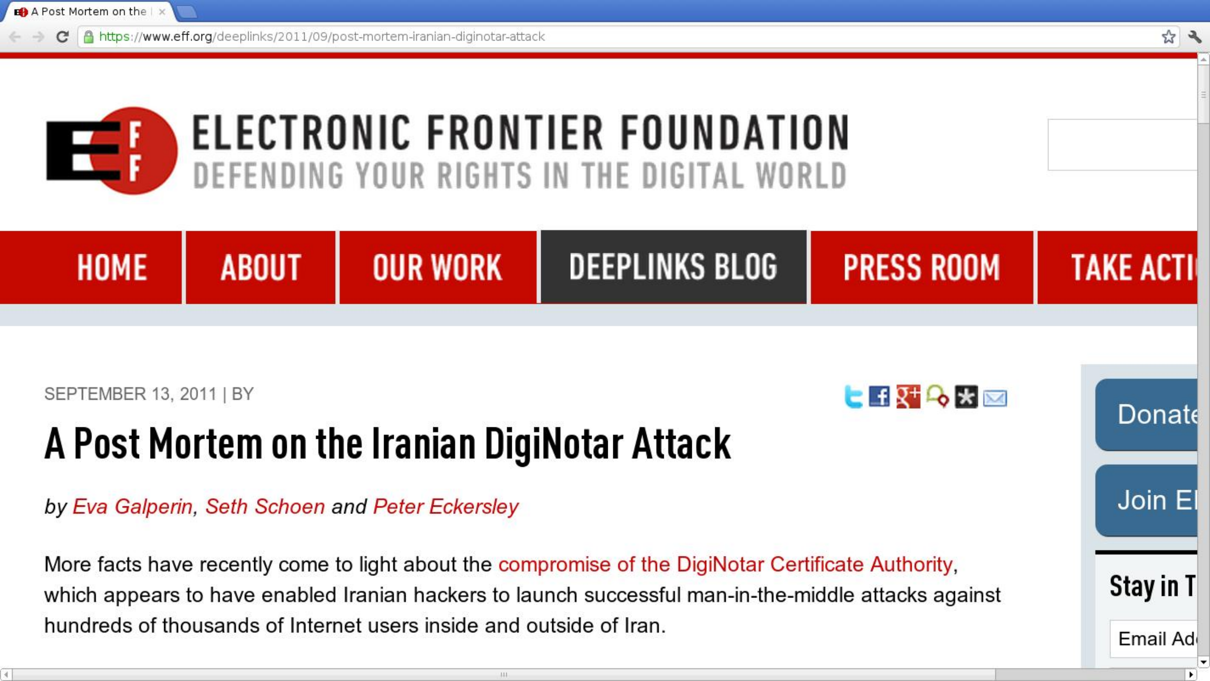
The story caused such a stir that the NSA's chief General Keith Alexander was called before Congress last week to testify about the project and categorically denied the facility will be used to spy on American citizens.

"The NSA does not have the ability to do that in the United States," Alexander told Georgia Rep. Hank Johnson. "We're not authorized to do that, nor do we have the equipment in the United States to collect that kind of information."

NSA public information officer Vanee' Vines backed up Alexander in an email saying: "What it will be is a state-of-the-art facility designed to support the Intelligence Community's efforts to further strengthen and protect the nation."

Update: The NSA does not spy on Americans, they hire it out to the Israelis.

While it's impossible to know the specifics of the work to be done in Bluffdale, it's pretty clear the NSA does have the power to snoop on Americans at will, despite what General Alexander said to Congress.

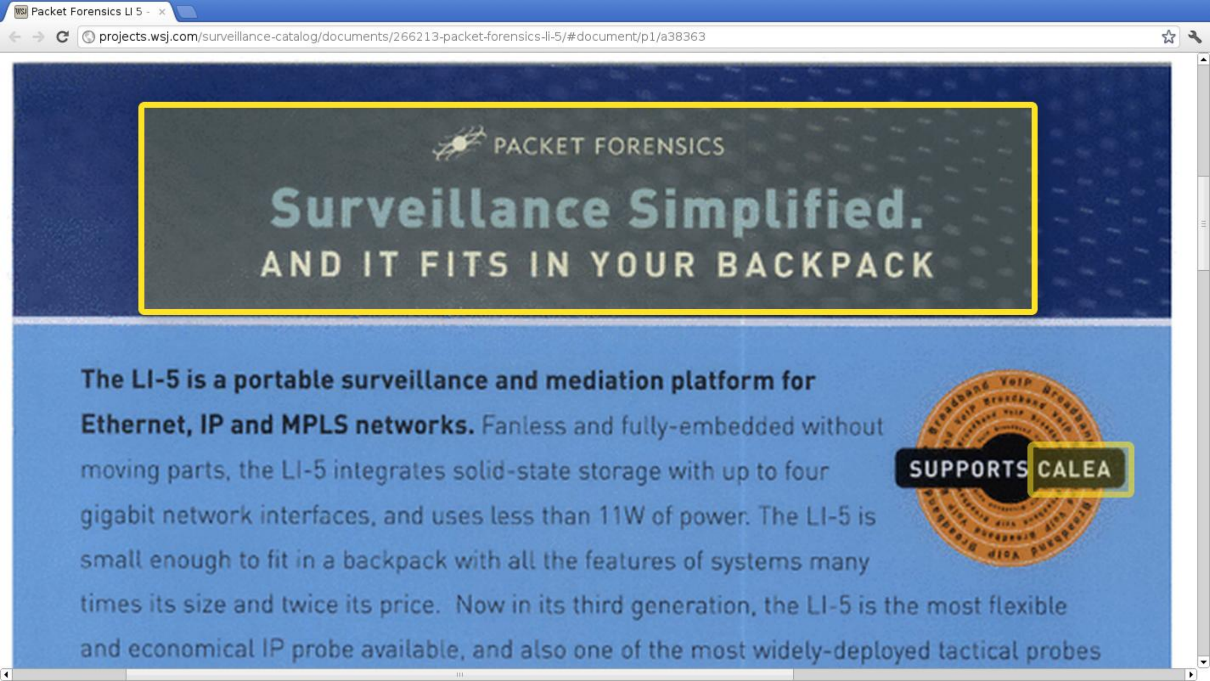

A Post Mortem on the I
https://www.eff.org/deeplinks/2011/09/post-mortem-iranian-diginotar-attack
ELECTRONIC FRONTIER FOUNDATION
DEFENDING YOUR RIGHTS IN THE DIGITAL WORLD
HOME
ABOUT
OUR WORK
DEEPLINKS BLOG
PRESS ROOM
TAKE ACTI
Donate
Join E
Stay in T
Email Ad

SEPTEMBER 13, 2011 | BY

A Post Mortem on the Iranian DigiNotar Attack

by Eva Galperin, Seth Schoen and Peter Eckersley

More facts have recently come to light about the compromise of the DigiNotar Certificate Authority, which appears to have enabled Iranian hackers to launch successful man-in-the-middle attacks against hundreds of thousands of Internet users inside and outside of Iran.


Packet Forensics LI 5 -
projects.wsj.com/surveillance-catalog/documents/266213-packet-forensics-li-5/#document/p1/a38363

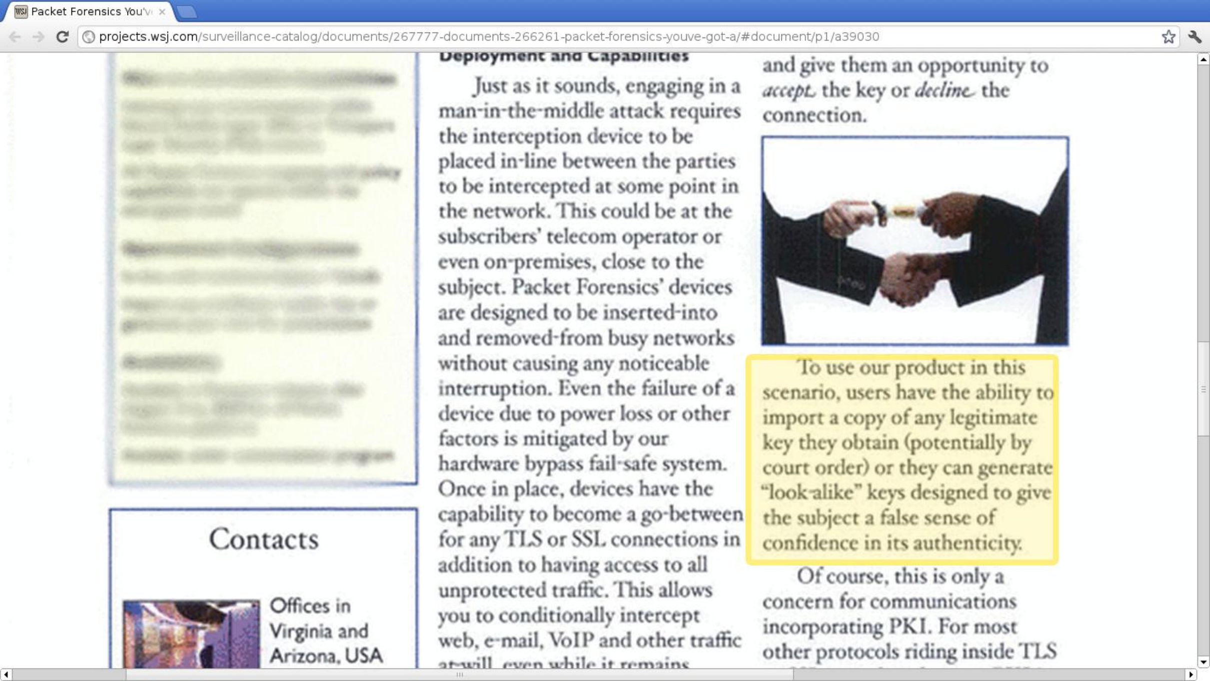
PACKET FORENSICS

Surveillance Simplified.

AND IT FITS IN YOUR BACKPACK

SUPPORTS CALEA

The LI-5 is a portable surveillance and mediation platform for Ethernet, IP and MPLS networks. Fanless and fully-embedded without moving parts, the LI-5 integrates solid-state storage with up to four gigabit network interfaces, and uses less than 11W of power. The LI-5 is small enough to fit in a backpack with all the features of systems many times its size and twice its price. Now in its third generation, the LI-5 is the most flexible and economical IP probe available, and also one of the most widely-deployed tactical probes


Packet Forensics You'v
projects.wsj.com/surveillance-catalog/documents/267777-documents-266261-packet-forensics-youve-got-a/#document/p1/a39030

[illegible]

Contacts

Offices in Virginia and Arizona, USA

Deployment and Capabilities

Just as it sounds, engaging in a man-in-the-middle attack requires the interception device to be placed in-line between the parties to be intercepted at some point in the network. This could be at the subscribers' telecom operator or even on-premises, close to the subject. Packet Forensics' devices are designed to be inserted-into and removed-from busy networks without causing any noticeable interruption. Even the failure of a device due to power loss or other factors is mitigated by our hardware bypass fail-safe system. Once in place, devices have the capability to become a go-between for any TLS or SSL connections in addition to having access to all unprotected traffic. This allows you to conditionally intercept web, e-mail, VoIP and other traffic at-will, even while it remains

and give them an opportunity to *accept* the key or *decline* the connection.

To use our product in this scenario, users have the ability to import a copy of any legitimate key they obtain (potentially by court order) or they can generate "look-alike" keys designed to give the subject a false sense of confidence in its authenticity.

Of course, this is only a concern for communications incorporating PKI. For most other protocols riding inside TLS

Why does this matter?

Most crypto isn't designed to resist serious attackers:

- Active forgeries break "opportunistic encryption" etc.
- Trusted third parties (e.g., CAs) are frequently compromised.
- General Michael Hayden: "We kill people based on metadata."
- Future quantum computers will break RSA, DSA, ECC.

Why does this matter?

Most crypto isn't designed to resist serious attackers:

- Active forgeries break "opportunistic encryption" etc.
- Trusted third parties (e.g., CAs) are frequently compromised.
- General Michael Hayden: "We kill people based on metadata."
- Future quantum computers will break RSA, DSA, ECC.

Academics have trouble demonstrating these attacks
⇒ incentive to write papers about other things.