Security Now! #471 - 09-02-14

PGP: Time for an upgrade?

This week on Security Now!

- The iCloud iBrute iHack... fun with naked celebrities
- The Russians are coming (with their mega password-cracking database)
- More consumer WiFi router security troubles
- "Stingray" Fake Cell Phone "Towers" (base-stations)
- Another CryptoLocker Clone in the wild.
- China's own Operating System
- SQLR progress report
- The new trouble with RAID 5
- Encrypting eMail ... with PGP??

The relative size of similar-security public keys:

---BEGIN PGP PUBLIC KEY BLOCK---
Version: GnuPG/MacOSX 2.0.20 (Darwin)

my miniLock ID 26rSAYFyGAvJoof52NhPFRf1uMC4HTlqC73k6J3k

-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG/MacOSX 2.0.20 (Darwin)

my miniLock ID 26rSAYFyGAvJoof52NhPFRf1uMC4HTlqC73k6J3k

-----END PGP PUBLIC KEY BLOCK-----
Security News:

The iCloud iBrute iHack

- Github / ibrute: HackApp (@hackappcom)
  - The end of fun, Apple have just patched
    Here is appleID password bruteforce pOc. It's only p0c, so there is no
    - MultiThreading feature
    - Save-State-On-Exception feature
  - do it yourself

It uses Find My Iphone service API, where brute force protection was not implemented. Password list was generated from top 500 RockYou leaked passwords, which satisfy appleID password policy. Before you start, make sure it's not illegal in your country.

Be good :)

Namecheap's Blog reports massive SUCCESSFUL attack

- [http://community.namecheap.com/blog/2014/09/01/urgent-security-warning-may-affec](http://community.namecheap.com/blog/2014/09/01/urgent-security-warning-may-affec)
- DNS, Hosting, SSL Guidance
- "We make registering, hosting, and managing domains for yourself or others easy and affordable, because the internet needs people."
- 1.2 Billion unique pairs of usernames and passwords
  - with 542 million unique eMail addresses... in the hands of Russians
  - The "CyberVors" based in Southern central Russia
  - 420,000 SQL injection vulnerable websites.

- Matt Russell, VP Hosting: (posted on Labor Day)
  Overnight, our intrusion detection systems alerted us to a much higher than normal load against our login systems. Upon investigation, we determined that the username and password data gathered from third party sites, likely the data identified by The Register (i.e. not Namecheap) is being used to try and gain access to Namecheap.com accounts.

  The group behind this is using the stored usernames and passwords to simulate a web browser login through fake browser software. This software simulates the actual login process a user would use if they are using Firefox/Safari/Chrome to access their Namecheap account. The hackers are going through their username/password list and trying each and every one to try and get into Namecheap user accounts.

  The vast majority of these login attempts have been unsuccessful as the data is incorrect or old and passwords have been changed. As a precaution, we are aggressively blocking the IP addresses that appear to be logging in with the stolen password data. We are also logging these IP addresses and will be exporting blocking rules across our
network to completely eliminate access to any Namecheap system or service, as well as making this data available to law enforcement.

While the vast majority of these logins are unsuccessful, some have been successful. To combat this, we’ve temporarily secured the Namecheap accounts that have been affected and are currently contacting customers involved requesting they improve the security for these accounts.

[...]

I must reiterate this is not a security breach at Namecheap, nor a hack against us. The hackers are using usernames and passwords being used have been obtained from other sources. These have not been obtained from Namecheap. But these usernames and passwords that the hackers now have are being used to try and login to Namecheap accounts.

Our early investigation shows that those users who use the same password for their Namecheap account that are used on other websites are the ones who are vulnerable.

**More WiFi Router Insecurity**

- WPS -- WiFi Protected Setup implementation errors.
- Bruce Schneier: "Attacks never get worse, they only ever get better."

- SN: Jan 10th, 2012, #335: "WiFi Protected (In)Security"
  - "Reaver" Stefan Viehbock, broke WPS:
    - "Brute forcing Wi-Fi Protected Setup: When poor design meets poor implementation."
    - WPS PIN is 8 digits (7 + checksum)
    - Possible to break it in half and check the first 4 and the second 3 separately! Whoops.
      - 10 Million in an online attack with wrong-guess timed lockout.
      - 10,000 + 1,000 tries = 11,000 tries... Whoops!
- WPS:
  - Aimed to allow easy connection.
  - Two primary modes: Pushbutton & PIN.
  - Provides the WPA passphrase to stations providing the proper PIN.

- Dominique Bongard, founder of Oxite, Switzerland
  - "It takes one second. It's nothing. Bang. Done."
  - Simple linear congruential PRNG with 32-bits of state.
  - The two crucial AES keys (E-S1 and E-S2) are generated immediately after the AP public handshake nonce.
So...

- Do the WPS protocol up to the third message
- Get the nonce provided in the 1st message
- Brute force the state of the PRNG
- Compute E-S1 and E-S2 from the state.
- Reverse the E-Hash1 and E-Hash2
- Brute force Pin1 and Pin2
- Restart the WPS protocol and obtain the AP's passphrase.

- Who's vulnerable:
  - The above is Broadcom... who used the insecure reference code for their routers.
  - A second unnamed vendor:
    - Linear Feedback Shift Register (LFSR)
    - The it always starts from zero... thus a PREDICTABLE pattern.
    - Trip the power, reboot the router, walk right into the network.
    - Vendor contacted and being given time to remediate.

  - (GRC's SQRL client's PRNG defeats ALL of those vulnerabilities.)

"StingRay" Fake cell phone towers
- Military Bases, Casinos
- <quote> The "VME Dominator", which is described as, “a real time GSM A5.1 cell phone interceptor. It cannot be detected. It allows interception of voice and text. It also allows voice manipulation, up or down channel blocking, text interpret and modification, calling & sending text on behalf of the user, and directional finding of a user during random monitoring of calls.”
- <quote> Pursuant to Federal law at 47 U.S.C. 302a, this product is available only for use by the Government of the United States or any agency thereof. Other interested parties are urged to contact appropriate regulatory oversight entities to determine whether any additional exceptions or arrangements have been authorized and implemented to permit use of this product consistent with controlling law.
- ACLU > Who's got'em?
  https://www.aclu.org/maps/stingray-tracking-devices-whos-got-them
- ACLU StingRay Info: https://www.aclu.org/node/37337

CryptoLocker Clone
- BleepingComputer's Lawrence Abrams, posted yesterday: (thanks @SimonZerafa)
- New CryptoLocker copycat ransomware in the wild
- Calls itself "CryptoLocker"... but it isn't.
- 1.8 BTC ($864)
- Uses "PHPSESSID" cookie to identity your "encryption session" ID. DON'T wipe cookies!
- Windows' Shadow Volume Copies are NOT deleted, so some recovery might be possible.

**China's own operating system**
- Would they not just adapt an entire Linux desktop environment?
- …or… Why reinvent the chopsticks?

**SQRL:**
- Outputting a QR code is WAY SIMPLER than inputting them!
- COM / DirectX / DirectShow

**SpinRite:**
- Why can't "SpinRite" be built into a drive?
- What about an OS?
- The end of RAID 5?… now we need RAID 6.
  - Unrecoverable error rates have grown large compared with drive size.
  - Rebuilding a RAID is the one time the RAID cannot sustain ANY errors.
  - But with drive sizes as large as they are, the chances are very good that a read error WILL occur!

---

**PGP and eMail Encryption**

**Matthew Green, "Cranky" Cryptographer at Johns Hopkins**


Matthew’s blog began with:
Last Thursday, Yahoo announced their plans to support end-to-end encryption using a fork of Google's end-to-end email extension. This is a Big Deal. With providers like Google and Yahoo onboard, email encryption is bound to get a big kick in the ass. This is something email badly needs.

So great work by Google and Yahoo! Which is why following complaint is going to seem awfully ungrateful. I realize this and I couldn't feel worse about it.

Concluded with:
I realize I sound a bit cranky about this stuff. But as they say: a PGP critic is just a PGP user who's actually used the software for a while. At this point so much potential in this area and so many opportunities to do better. It's time for us to adopt those ideas and stop looking backwards.
First of all:
- PGP was/is a beautiful piece of design work by Phil Zimmerman

Issues:
- S/MIME has the benefit of being integrated into eMail. PGP rides along in plain sight on top.
  - This is probably more acceptable when the Internet was for nerds.
- "PGP keys suck":
  - [Image](http://4.bp.blogspot.com/-5p2Kbnp54WE/U-jqIUldwHI/AAAAAAAABA4/Y1t0c9YOa8/s1600/keys.png)
  - All three keys offer approximately the same level of security.
- "Managing Keys sucks"
  - Key servers, Webs of Trust, Signing parties, Public Key Fingerprints
  - (S/MIME uses the traditional PKI with CAs or self-signed keys.)
  - iMessage manages keys... but with convenience comes loss of control.
- Lack of any Forward Security:
  - If keys are ever compromised, all past (perhaps archived) encrypted messages can be decrypted.
  - Online protocols like negotiating a real-time connection can provide this easily, for Offline protocols -- like eMail encryption keying -- it's not impossible, but it's much more involved.

So what SHOULD we be doing?
- A proper approach to key management. This could be anything from centralized key management as in Apple's iMessage -- which would still be better than nothing -- to a decentralized (but still usable) approach like the one offered by Signal or OTR. Whatever the solution, in order to achieve mass deployment, keys need to be made much more manageable or else submerged from the user altogether.
- Forward secrecy baked into the protocol. This should be a pre-condition to any secure messaging system.
- Cryptography that post-dates the Fresh Prince. Enough said.
- Screw backwards compatibility. Securing both encrypted and unencrypted email is too hard. We need dedicated networks that handle this from the start.

What's coming:
- Trevor Perrin's "Axolotl ratchet"
  - [GitHub](https://github.com/trevp/axolotl/wiki)
- SCIMP (Silent Circle Instant Messaging Protocol)
  - [Silent Circle](https://silentcircle.com/scimp-protocol)
DarkMail:
  - https://www.darkmail.info/
  - Phil Zimmerman, Ladar Levinson, et al
  - "Silent Circle and Lavabit are developing a new way to do email with end-to-end encryption. We welcome like-minded organizations to join our alliance."

Adam Caudill's "SMIMP" (Simple Messaging and Identity Management Protocol)
  - https://github.com/smimp/smimp_spec/blob/master/smimp_specification.md
  - Our friend Taylor Hornby (aka "FireXware") has been involved in that effort.
  - "SMIMP is a communication and identity system designed to address the modern threats that weren’t considered when the traditional email system was designed. Transparent encryption, forward secrecy, simple self hosting, auditable user information, and strong privacy are all baked into the design from the beginning."

MailPile:
  - https://www.mailpile.is/
  - at Alpha II level -- build upon OpenPGP.