SECDSM

Crypto Update 2016

Thx to Brandon

- For getting this thing off the ground, coordinating and facilitating this group.
- Thx to everyone else for supporting, presenting, attending.

Agenda

- Introduction/Disclaimer
- Opening Thoughts
- Crypto in the News
- In-transit
- At-rest
- Questions to ask vendors and colleagues

Opening Thoughts

- *Cryptography will not be broken, it will be bypassed* - Adi Shamir (the S in RSA)
- Never ever roll your own
- Randomness should not be left to chance
- Fail closed or degrade gracefully
- Generate and manage keys with care
- Encoding != Encryption. If there isn't a key, it's not encryption
- RIP SSL
- Kerckhoff's Principle
- CloudFlare: *"it's important to be able to understand the technology behind any security system in order to trust it"*
- This is hard

Crypto In The News

- Juniper Networks and ECC implant
- Cisco ASA IKEv1/2 Buffer Overflow
- DROWN
- Apple and FBI - no slide needed, just this comment from Alex Stamos (Facebook CSO): *“That it’s like drilling a hole in the windshield”*
- NSA Suite B Changes (and Post-Quantum Crypto)
- Ransomware…..

Juniper Networks

- Dual EC - NIST + NSA.
- Router/VPN appliance - 2008 changed to use Dual EC including additional changes to exploit via single VPN handshake and passively decrypt traffic
- 2012 - Juniper attacked and source code modified; only 1 constant for Dual EC changed to now allow adversaries to have access. Deployed via system updates and not discovered until 2016.
- OPM used Juniper devices
- http://blog.cryptographyengineering.com/2015/01/hopefully-last-post-ill-ever-write-on.html

Cisco ASA IKEv1/2

- This could have been big - UDP packets, a buffer overflow and possibility of a root shell. It was the reason I was late to the inaugural meeting.
- CVSSv2 Score of 10
- How did this bug get past the army of coders and what about ASLR and DEP mitigations on the one of the world's most popular security appliance?
- What if someone owned your VPN/Firewall/IPS chassis?

DROWN

Decrypting RSA using Obsolete and Weakened eNcryption

- Considered a cross-protocol attack - OpenSSL (server-side) may still honor SSLv2 cipher suite requests despite the disabled configuration option.
- SSLv2 and TLS use a form of RSA encryption padding (PKCS#1 v1.5) which does not fail closed (Chosen Ciphertext Attack). Padding has been improved in TLS (OAEP) but not in SSL.
- Why does this matter? No Pre-Master Secret in SSL…Master Secret can be as short as 40 bits and used with weak export ciphers leading to recovery?
- Then what - well, do you use different certificates (private keys) for SSL vs. TLS?
    - *“The "general DROWN" attack actually requires watching about 1,000 TLS handshakes to find a vulnerable RSA ciphertext, about 40,000 queries to the server, and about 2^50 offline operations.”*
    - *“While the attack described above seems costly, it can be conducted in several hours and $440 on Amazon EC2. Are your banking credentials worth $440? Probably not. But someone else's probably are. Given all the things we have riding on TLS, it's better for it not to be broken at all.”*
- http://blog.cryptographyengineering.com/2016/03/attack-of-week-drown.html
- https://drownattack.com/drown-attack-paper.pdf

NSA Suite B

- NIST+NSA Suite B has been the gold standard for sometime.
    - Wickr used/uses Suite B crypto
- About 6 months ago, NSA issued new guidance on Suite B related to post-quantum cryptography
- New suite not chosen - and some guidance to not move to EC crypto?
- AES, EC (DH key exchange and DSA), SHA and RSA with strong keys.
- Quantum computing still in infancy (as best we can tell)
- Quantum algorithms exist but if they can be built remains to be seen. Quantum exhaustive search can yield some info.
    - Grover's algorithm - essentially halves the key length for AES
    - Shor's algorithm (1994) for prime number factorization is exponential speed gain which is bad news for public key crypto
- http://pqcrypto.org/www.springer.com/cda/content/document/cda_downloaddocument/9783540887010-c1.pdf
- http://blog.cryptographyengineering.com/2012/04/its-end-of-world-as-we-know-it-and-i.html
- https://www.nsa.gov/ia/programs/suiteb_cryptography/

Ransomware

- Even these jerks get it wrong…
- Linux.Encoder.1 is file encrypting ransomware that targets Linux operating systems.
- “*rather than generating secure random keys and IVs, the sample would derive these two pieces of information from the libc rand() function seeded with the current system timestamp at the moment of encryption. This information can be easily retrieved by looking at the file’s timestamp”*
- https://labs.bitdefender.com/2015/11/linux-ransomware-debut-fails-on-predictable-encryption-key/

In-Transit

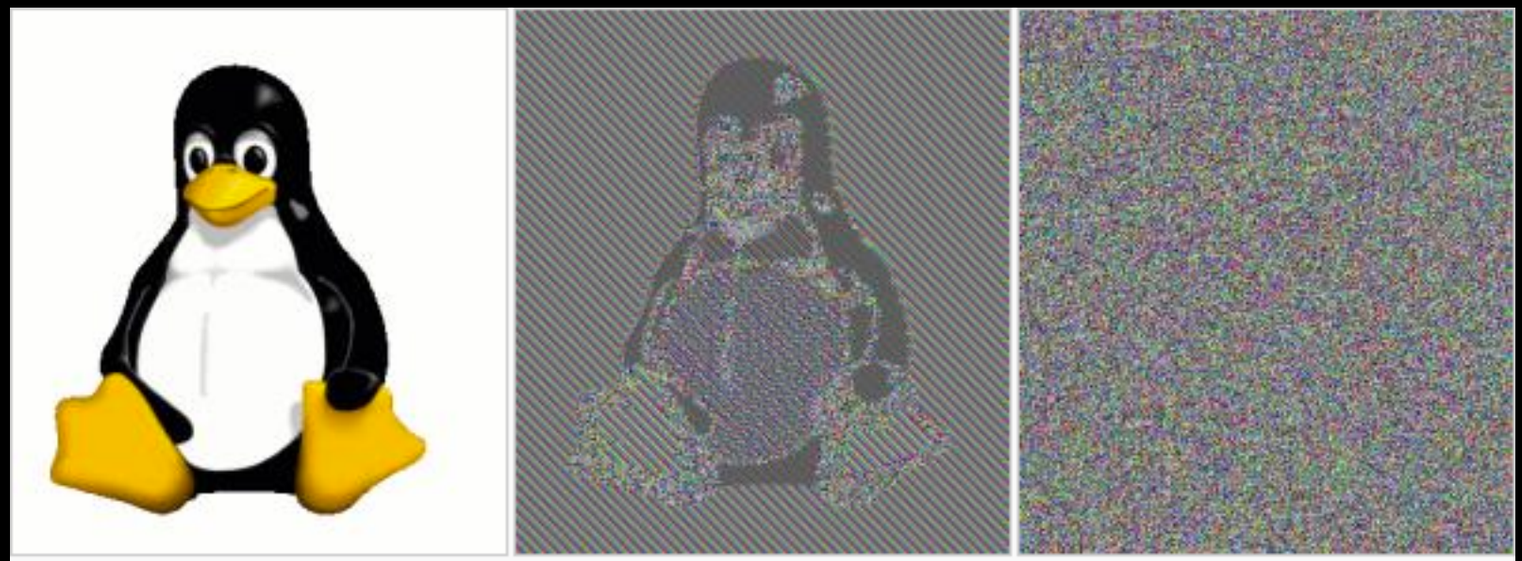
- Cryptographic Right Answers courtesy of Tom Ptacek: https://gist.github.com/tqbf/be58d2d39690c3b366ad
- TLS for client-server application security. OpenSSL, BoringSSL, Amazon ELBs and perhaps a secure edge provider like CloudFlare
  - https://blog.cloudflare.com/do-the-chacha-better-mobile-performance-with-cryptography/
- Cipher Suites may be enhanced for devices without AES-NI/cryptographic extensions: ChaCha10-Poly1305
- Enable HSTS
- Check yourself: ssllabs.com

At-Rest - Encryption

- Cryptographic Right Answers courtesy of Tom Ptacek: https://gist.github.com/tqbf/be58d2d39690c3b366ad
- Authenticated Encryption expected in 2016 (Chosen Plaintext Attack (CPA)+ cipher text integrity)
- AES-GCM mode (Galois Counter Mode) is fast, hardware-accelerated and provides a Message Authentication Code (MAC) for each block of cipher text and is nonce-based.
  - Encrypt then MAC
  - Counter Mode allows for parallelization
- Other options (if you are willing to stray from NIST): NaCL, ChaCha-Poly1305

At-Rest - Encryption

- Some pitfalls:



Original image | Encrypted using ECB mode | Modes other than ECB result in pseudo-randomness

- AES-CBC is quite common but has some opportunities (not Authenticated Encryption, Padding Oracle attacks, parallelization challenges)
  - AES-CBC with a blank IV is probably the same as ECB
  - pycrypto (AppEngine) doesn't support GCM
- ***Never, ever use ECB mode of AES.***
- http://www.owenstephens.co.uk/programming/2009/10/21/aes-using-ecb-demo-using-python.html

At-Rest - Encryption

- Randomization super important (remember jerks from Linux.Encoder.1)
- Tom Ptacek: ***Use urandom. Use urandom. Use urandom. Use urandom. Use urandom. Use urandom.***
- /dev/urandom is a kernel-space CSPRNG (raw device entropy) which does not block
- /dev/random is based on the same CSPRNG as /dev/urandom but it doesn't block which is pretty darn important for something like Docker or busy instances/servers.
- JRE: java.security.SecureRandom - be careful with getInstanceStrong blocking
    - https://docs.oracle.com/javase/8/docs/api/java/security/SecureRandom.html
- .NET: CryptGenRandom, System.Security.Cryptography, not System.Random
    - https://msdn.microsoft.com/en-us/library/system.security.cryptography.rngcryptoserviceprovider.aspx
- http://sockpuppet.org/blog/2014/02/25/safely-generate-random-numbers/
- http://www.2uo.de/myths-about-urandom/
- https://speakerdeck.com/filosottile/the-plain-simple-reality-of-entropy-at-32c3

At-Rest - Hashing

- If you intend to only compare data, consider cryptographic hashing.
- SHA-2 family or Password-based Hashing. SHA-3 is now a NIST standard (August 2015): http://www.nist.gov/itl/csd/sha-100212.cfm
- PBKDF2 (massive rounds), scrypt or bcrypt with appropriate work factors
- Salts (global or unique based on your lookup scheme)
- Pepper: extra fixed salt or salt “range”
- Facebook scheme: https://twitter.com/FiloSottile/status/552830697942319105

Questions (Vendors, colleagues, friends, family, etc)

- How are keys managed (generated, rotated and data re-keyed)?
- Who has access to the keys?
- What block or stream ciphers are used including modes?
- Does the solution leverage Authenticated Encryption/Authenticated Encryption with Authenticated Data (AEAD)?
- Does the solution provide End-to-End Encryption?
- How does the solution deal with downgrade attacks?
- Is there any custom or proprietary cryptography?
- Does the solution protect against Chosen Plaintext Attacks (CPA)?
- Does the solution protect against Chosen Ciphertext Attacks (CCA)?
- Does the solution use libraries with known cryptographic vulnerabilities (ex: non-constant time algorithms)?

Further Info/Reading/ Knowledge

- https://crypto.stanford.edu/~dabo/cryptobook/draft_0_2.pdf
- https://blog.lastpass.com/2015/06/lastpass-security-notice.html/
- Coursera Crypto Courses