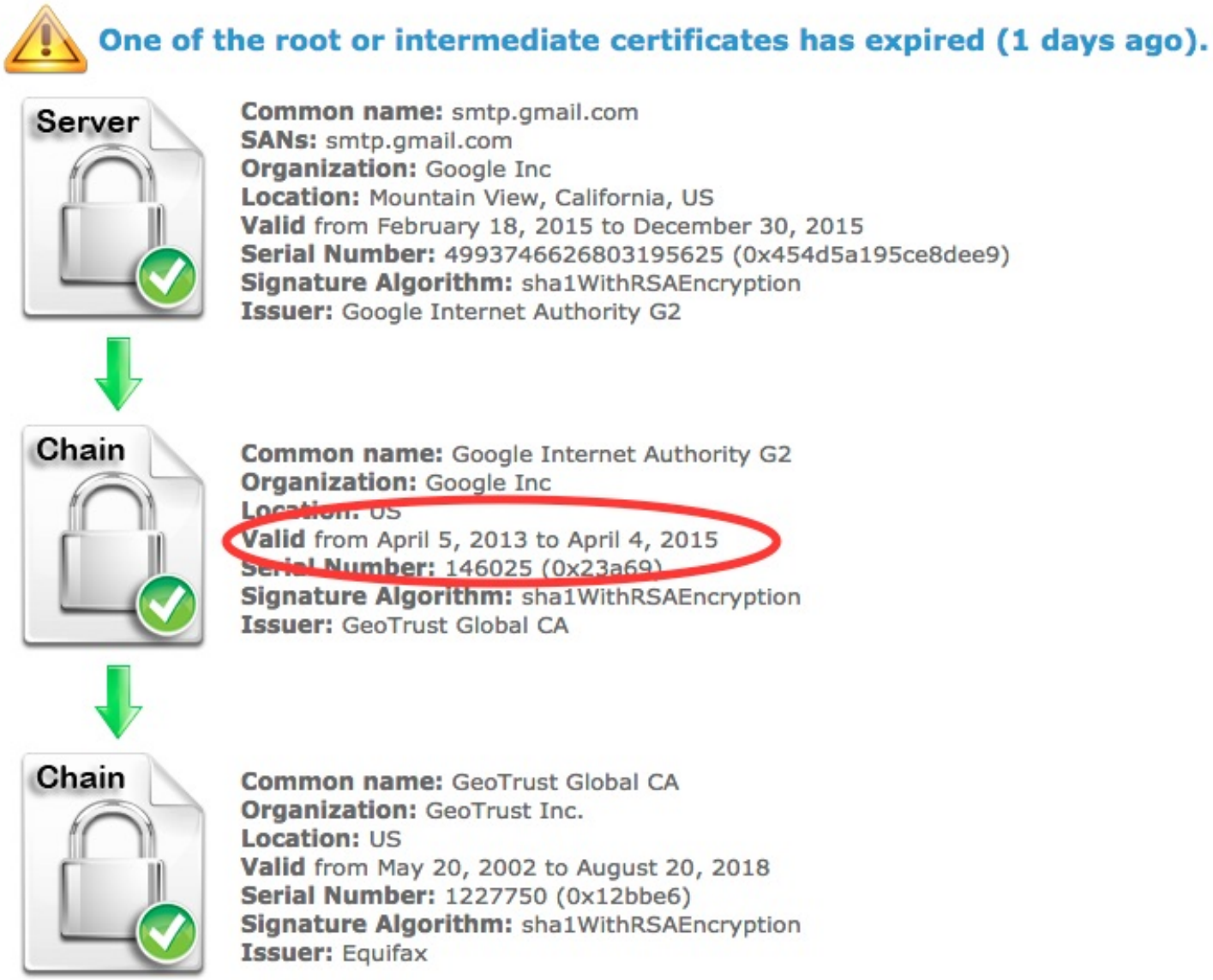
Security Now! #502 - 04-07-15

The TrueCrypt Audit

(Elaine is fine! -- It was apparently just a transient eMail glitch. All is wel!)

This week on Security Now!

- The ten thousand pound giant shifts its weight: Google REVOKES CNNIC's Root Cert!
- Firefox briefly offered "Opportunistic Encryption" of HTTP connections.
- A key Google certificate expired and is replaced with... "Obsolete Cryptography."
- Microsoft to change handling of Do Not Track.
- Predictably, The IPv4 address space market is heating up.
- A close look at the finding of the TrueCrypt audit!



Security News

CNNIC's Root CA cert to be removed from Chrome:

- http://googleonlinesecurity.blogspot.com/2015/03/maintaining-digital-certificate-security.html
- Recap:
  - On Friday, March 20th, Google was alerted to the attempted use of fraudulent digital certificates for several Google domains.
  - The certificates were issued by an intermediate certificate authority apparently held by a company called MCS Holdings.
  - This intermediate certificate was issued by CNNIC.

- April Fools Day -- but this was no joke!
  - <<Adam Langley writes>>: As a result of a joint investigation of the events surrounding this incident by Google and CNNIC, we have decided that the CNNIC Root and EV CAs will no longer be recognized in Google products. This will take effect in a future Chrome update. To assist customers affected by this decision, for a limited time we will allow CNNIC's existing certificates to continue to be marked as trusted in Chrome, through the use of a publicly disclosed whitelist. While neither we nor CNNIC believe any further unauthorized digital certificates have been issued, nor do we believe the misissued certificates were used outside the limited scope of MCS Holdings' test network, CNNIC will be working to prevent any future incidents. CNNIC will implement Certificate Transparency for all of their certificates prior to any request for reinclusion. We applaud CNNIC on their proactive steps, and welcome them to reapply once suitable technical and procedural controls are in place.

- Are we all just one big happy family?? (Maybe something was lost in translation?)
  - http://www1.cnnic.cn/AU/MediaC/Announcement/201504/t20150402_52049.htm
  - The decision that Google has made is unacceptable and unintelligible to CNNIC, and meanwhile CNNIC sincerely urge that Google would take users' rights and interests into full consideration.
  - For the users that CNNIC has already issued the certificates to, we guarantee that your lawful rights and interests will not be affected.
  - China Internet Network Information Center(CNNIC) / April 2nd, 2015

FireFox briefly offered Opportunistic Encryption (OE) TLS

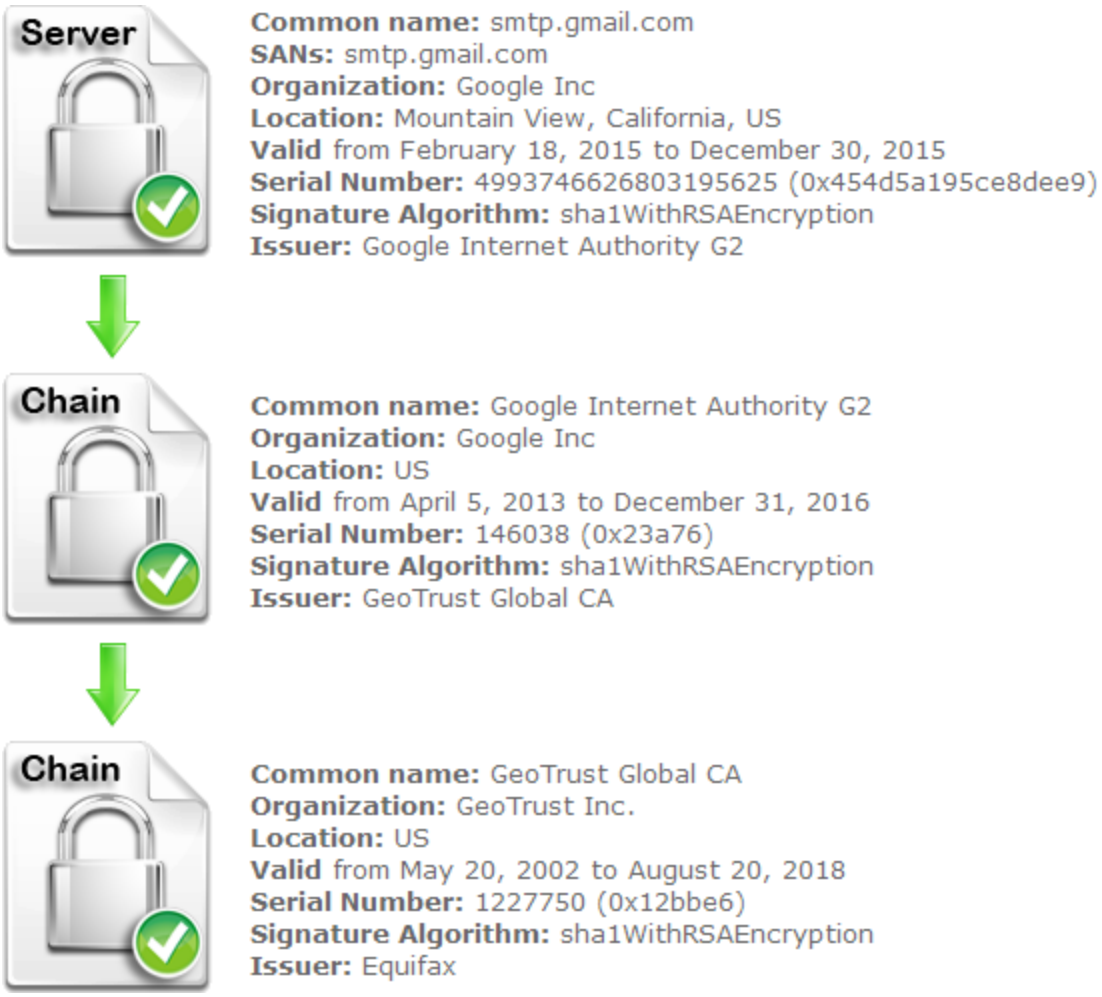
- We had it briefly in v37... but then it was taken away by v37.0.1
- Offers the opportunity for encrypted http:// content.
- How:
  - Install a TLS based http/2 or SPDY server on a separate port... like 443.
  - That server's certificate can be self-signed since OE is non-authenticated.
  - Add a response header Alt-Svc: h2=":443" or spdy/3.1
  - (If you are using a spdy enabled server like nginx.)

- What:
  - When the browser encounters that response header it will start to verify that there is an HTTP/2 service on port 443 of the same domain.
  - When a session with that port is established, Firefox start routing the requests it would normally send in-the-clear to port 80 onto port 443... with encryption.
  - There will be no delay in responsiveness because the new connection is fully established in the background before being used. Firefox simple begins using the alternative connection once it becomes available.
  - If the alternative service (port 443) becomes unavailable or cannot be verified, Firefox will automatically return to using cleartext on port 80.
  - Other clients that don't recognize the Alt-Svc: header ignore it and continue using port 80.
- Note:
  - The fail-back to port 80 is TRIVIAL for any attacker to cause, so this really is only useful for purely PASSIVE network eavesdropping protection.
  - But to thwart the bulk collection of otherwise in-the-clear text it's better than nothing.
  - OE won't work with HTTP/1 servers because /1 protocol does not include the "scheme" as part of each transaction which is needed for the Alt-Svc: solution.
- Also:
  - This Alt-Svc: mapping is RETAINED and used in the future.
- What happened with v37.0.1?
  - https://www.mozilla.org/en-US/security/advisories/mfsa2015-44/
  - An asture security researcher discovered a flaw in Mozilla's HTTP Alternative Services implementation. If an Alt-Svc header is specified in the HTTP/2 response, SSL certificate verification can be bypassed for the specified alternate server. As a result of this, warnings of invalid SSL certificates will not be displayed and an attacker could potentially impersonate another site through a man-in-the-middle (MTIM), replacing the original certificate with their own.
- RFC:
  - https://tools.ietf.org/html/draft-ietf-httpbis-alt-svc-04

Google's smtp.google.com cert CHAIN had trouble

- Last Saturday morning, the INTERMEDIATE certificate chained into the "smtp.google.com" certificate expired.
- That server is used by Gmail and Google Apps users to send outgoing mail.
- The certificate was issued by the intermediate CA Google Internet Certificate Authority G2 which issues certificates for Google web sites and other properties.
- A look at the NEW "smtp.google.com" certificate:
- https://www.sslshopper.com/ssl-checker.html?host=smtp.google.com#hostname=smtp.gmail.com:465

A look at the new certificate chain:



- "Your connection to www.grc.com is encrypted with obsolete cryptography."

Microsoft to change handling of Do Not Track

- http://blogs.microsoft.com/on-the-issues/2015/04/03/an-update-on-microsofts-approach-to-do-not-track/
- <<quote>> [April 3rd]
  As industry standards evolve, how we implement those standards evolve as well. So to reflect the current requirements of the privacy standard for tracking preferences, Microsoft is changing how Do Not Track (DNT) is implemented in future versions of our browsers: We will no longer enable it as the default state in Windows Express Settings.

  While our implementation of DNT two years ago in Internet Explorer 10 was welcomed by many, others voiced concerns, especially given that discussions were underway at the time to establish an industry wide standard for user tracking preferences.

  Since then, the World Wide Web Consortium (W3C) has continued to refine language to address how users express a preference regarding tracking. The latest draft of the

standard reads:

Key to that notion of expression is that the signal sent MUST reflect the user's preference, not the choice of some vendor, institution, site, or network-imposed mechanism outside the user's control; this applies equally to both the general preference and exceptions. The basic principle is that a tracking preference expression is only transmitted when it reflects a deliberate choice by the user. ***In the absence of user choice, there is no tracking preference expressed.*** *(Emphasis added.)*

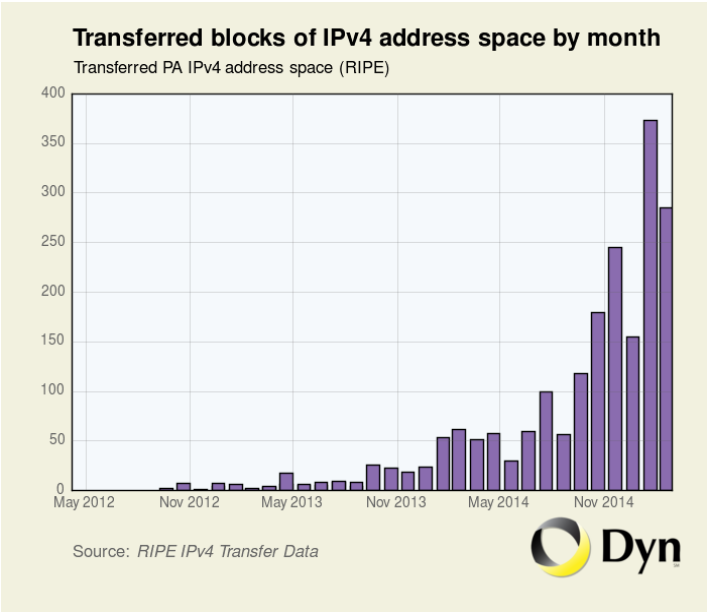
Put simply, we are updating our approach to DNT to eliminate any misunderstanding about whether our chosen implementation will comply with the W3C standard. Without this change, websites that receive a DNT signal from the new browsers could argue that it doesn't reflect the users' preference, and therefore, choose not to honor it.

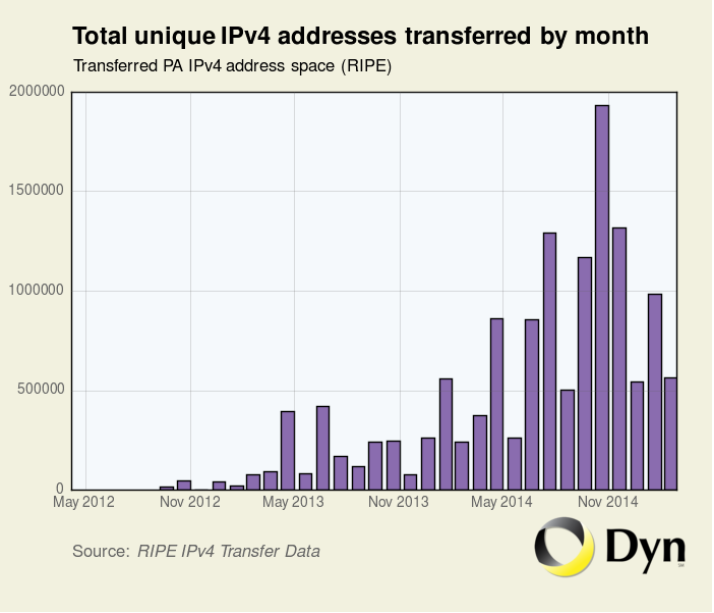
As a result, DNT will not be the default state in Windows Express Settings moving forward, but we will provide customers with clear information on how to turn this feature on in the browser settings should they wish to do so. This change will apply when customers set up a new PC for the first time, as well as when they upgrade from a previous version of Windows or Internet Explorer.

We said in 2012 that browser vendors should clearly communicate to consumers whether the DNT signal is turned off or on, and make it easy for them to change the setting. We did that for IE 10 and IE 11. And we're continuing to do so with future versions of our browsers.

The "After Market" for IPv4 address space is heating up:

- http://research.dyn.com/2015/04/ipv4-address-market-takes-off/





- RIPE's table of transfers of "Provider Aggregatable" (PA) IPv4 address shows a rapidly increasing rate of transfers of IPv4 address blocks and unique IPv4 addresses.
- February 2015 saw 373 organizational transfers of major blocks
- November 2014 saw nearly 2 million unique addresses transferred.
- Romania appears to have a surplus and is cashing in:
  - 51% of ALL IPv4 blocks transferred came from a single Romanian organization, "Jump.ro."
  - Jump.ro will sell large blocks of IPv4 space for $10/address or lease smaller blocks for $0.50/address/year.
- Saudi Telecom received 1.5 million IP addresses in 17 block transfers. 14 blocks from Romania and 3 from Ukraine.... so at $10 each... $15 million dollars.

Miscellany

Firefox Configuration Bonanza!

- "Configuration Mania"
- https://addons.mozilla.org/en-us/firefox/addon/configuration-mania-4420/

The [fruitless] search for extraterrestrial intelligence?

- Barry Wallis @BarryWallis
  @SGgrc Aliens are smart, so of course all their comms are encrypted and indistinguishable from noise… Just exactly like what SETI sees everyday!

Telomere Length Measurement

- Steve Gibson: @SGgrc
- Curious about the average length of your telomeres?
- I am: http://bit.ly/1FdkSVU I've signed up. We'll see. :)
- https://www.indiegogo.com/projects/titanovo-measure-your-health/x/10426366

SpinRite:

Security Now listener Paul Windham recounts his first encounter with SpinRite...

Dear Steve,

As a contemporary SpinRite user, I have to tell you about an experience it had with SpinRite many years ago: I was in the Marine Corps in 29 Palms California in the 1980's. We had a PC which was running some kind of MSDOS and it would no longer boot. I was fortunate enough to work with a retired marine name James Kornegay. He help me all the time with PC issues and he owned a copy of SpinRite which we ran on the unbootable system... after which, of course, it booted right up and ran.

To this day, I am a SpinRite user, and it has saved me from a lot of pain.
Thanks for 500 shows and look forward to odometer reading 512.

Paul Windham

Spinrite user / Security now user.

The TrueCrypt Audit

Matthew Green's Summary:
http://blog.cryptographyengineering.com/2015/04/truecrypt-report.html?m=1

<Matt:> The TL;DR is that based on this audit, Truecrypt appears to be a relatively well-designed piece of crypto software. The NCC audit found no evidence of deliberate backdoors, or any severe design flaws that will make the software insecure in most instances.

That doesn't mean Truecrypt is perfect. The auditors did find a few glitches and some incautious programming -- leading to a couple of issues that could, in the right circumstances, cause Truecrypt to give less assurance than we'd like it to.

For example: the most significant issue in the Truecrypt report is a finding related to the Windows version of Truecrypt's random number generator (RNG), which is responsible for generating the keys that encrypt Truecrypt volumes. This is an important piece of code, since a predictable RNG can spell disaster for the security of everything else in the system.

The Truecrypt developers implemented their RNG based on a 1998 design by Peter Guttman that uses an entropy pool to collect 'unpredictable' values from various sources in the system, including the Windows Crypto API itself. A problem in Truecrypt is that in some extremely rare circumstances, the Crypto API can fail to properly initialize. When this happens, Truecrypt should barf and catch fire. Instead it silently accepts this failure and continues to generate keys.

This is not the end of the world, since the likelihood of such a failure is extremely low. Moreover, even if the Windows Crypto API does fail on your system, Truecrypt still collects entropy from sources such as system pointers and mouse movements. These alternatives are probably good enough to protect you. But it's a bad design and should certainly be fixed in any Truecrypt forks.

In addition to the RNG issues, the NCC auditors also noted some concerns about the resilience of Truecrypt's AES code to cache timing attacks. This is probably not a concern unless you're perform encryption and decryption on a shared machine, or in an environment where the attacker can run code on your system (e.g., in a sandbox, or potentially in the browser). Still, this points the way to future hardening of any projects that use Truecrypt as a base.

Truecrypt is a really unique piece of software. The loss of Truecrypt's developers is keenly felt by a number of people who rely on full disk encryption to protect their data. With luck, the code will be carried on by others. We're hopeful that this review will provide some additional confidence in the code they're starting with.

The Audit Details

- https://opencryptoaudit.org/
- https://opencryptoaudit.org/reports/TrueCrypt_Phase_II_NCC_OCAP_final.pdf

1.3 Findings Summary:

During the engagement, CS (Cryptography Services) identified four (4) issues, and none led to a complete bypass of confidentiality in common usage scenarios. The standard workflow of creating a volume and making use of it was reviewed, and no significant flaws were found that would impact it.

The most severe finding relates to the use of the Windows API to generate random numbers for master encryption key material among other things. While CS believes these calls will succeed in all normal scenarios, at least one unusual scenario would cause the calls to fail and rely on poor sources of entropy; it is unclear in what additional situations they may fail.

Additionally, CS identified that volume header decryption relies on improper integrity checks to detect tampering, and that the method of mixing the entropy of keyfiles was not cryptographically sound.

Finally, CS identified several included AES implementations that may be vulnerable to cache-timing attacks. The most straightforward way to exploit this would be using native code, potentially delivered through NaCl in Chrome; however, the simplest method of exploitation through that attack vector was recently closed off.

3.2 Vulnerabilities:

1. CryptAcquireContext may silently fail in unusual scenarios
    a. High Severity
2. AES implementation susceptible to cache-timing attacks
    a. High Severity
3. Keyfile mixing is not cryptographically sound Cryptography
    a. Low
4. Unauthenticated ciphertext in volume headers Cryptography
    a. Undetermined

"CryptAcquireContext may silently fail in unusual scenarios"
Testing on Windows XP indicates that if this is the first time a user has issued a call with the NULL container (parameter 2), the first call to CryptAcquireContext will fail, while the second, initializing a new KeySet, will succeed. A later version of Windows tested appears to succeed on the first call, but this was not thoroughly tested.

While disturbing, this issue should not cause failure on common Windows XP uses. However, this is not the correct method of calling CryptAcquireContext and it may cause failure on uncommon Windows configurations (spanning XP through Windows 8.1).

"Exploit Scenario"
"A user creates a TrueCrypt Volume on a company-managed machine. Because of the Group Policy Settings in place at the organization, TrueCrypt is unable to open a handle to a Cryptographic Service Provider, and falls back to insecure sources of randomness, potentially enabling brute-force attacks on the master key.

"As detailed in finding 1 on page 13, under certain conditions the Random Number Generator on Windows will not get cryptographically secure random data as an input to components such as master key generation. If this occurs, the random pool will instead be fed by:

- 11 pointers to a variety of application structures. These are 32-bit values, but have significantly more predictable structure than a random 32-bit value due to program layout.
- Process and thread IDs
- Milliseconds since Windows started
- Process startup time
- Cursor position
- Time of last input message
- Flags indicating what types of messages are in the queue
- The X & Y coordinate of the input caret and mouse cursor
- Statistics regarding the current memory usage13 and working set14 such as load (measured between 0 and 100), total physical memory, available virtual memory, and minimum and maximum working size
- Creation, User and Kernel execution time of the current thread and process
- Network Management Data
- Physical hard drive performance statistics

"AES implementation susceptible to cache-timing attacks" (Severity: High, Difficulty: High)

- AES performance optimizations depend upon large pre-computed lookup tables.
- These tables do not comfortably fit entirely within CPU cache (where they would enjoy equal access speed.)
- The pattern of table access is dependent upon the data being encrypted and the secret key material.
- Therefore... if attackers can both arrange to feed their own plaintext data through the cipher system while carefully measuring the system's performance to detect processor cache hits and misses, secret keying material can potentially be obtained.

- "Exploit Scenario"
    - An attacker may be able to extract AES keys used to protect encrypted volumes. A successful exploit may rely on the attacker's ability to execute native code on the victim's machine, but recent advances in cache attacks performed by untrusted JavaScript indicate this area is being researched more heavily.
    - **BUT... this *ONLY* works if the TrueCrypt volume is MOUNTED and already accessible to an attacker.**

"Keyfile mixing is not cryptographically sound" (Severity: Low, Difficulty: High)

Description: TrueCrypt allows the use of Keyfiles that are included with the user's passphrase in the derivation of the key used to unlock a volume. However, TrueCrypt does not mix the keyfile content into the passphrase in a cryptographically sound manner.

A 64-byte buffer is constructed, initially zero, called the keypool that is used to hold the entropy generated from the keyfiles. For each keyfile, a maximum of 1024 Kilobytes are read. A CRC (initially 0xFFFFFFFF and using the polynomial 0x04c11db7) is constructed, and for each byte in the file it is updated. Each time the CRC is updated, its four bytes are individually added into the keypool, modulo 256, and advancing (so the first time it updates bytes 0-3, the second time 3-7, and so on, wrapping around when it reaches 64.) The keypool output at the end of the first keyfile is used as the input keypool for the second keyfile.

After all of the keyfiles are processed, each keypool byte is added (modulo 256) into the user's password byte at that position. If the password is less than 64 bytes, the keypool byte in that position is used directly.

The use of CRC in this way is not cryptographically sound. When mixing entropy from multiple sources, an attacker who controls one source of entropy should not be able to fully negate or manipulate the other sources, even if the attacker is aware of what the other data is. The use of a cryptographic hash function is the correct way to mix entropy together – assuming the hash function is unbroken, the best attack able to be mounted is a brute-force search for an input that, when combined with the uncontrolled input, yields a desirable output.

(No Exploit Scenario)... because there really isn't one.

Bad guy present WHILE a TrueCrypt volume is being generated.

"Unauthenticated ciphertext in volume headers" (Severity: Undetermined, Difficulty: High)

- The volume header and the volume are encrypted separately.
- The header contains the volume's master key.
- The header is encrypted with a key derived from the user-supplied password.
- TrueCrypt DOES provide detection of header corruption:
    - A magic string "TRUE" at the beginning of the volume header.
    - A CRC32 calculated over the master key material.
    - A CRC32 calculated over the remainder of the volume header.