System-wide crypto policies what and why

Tomáš Mráz
Crypto team
2019-01-26
AGENDA

• Introduction
• Cryptography and time
• Perspectives
• Crypto-policies - What is it?
• Crypto-policies - Why do we need it?
• Some demo
• Some details
• Crypto-policies What it isn't
• Future
INTRODUCTION
CRYPTOGRAPHY AND TIME
CRYPTOGRAPHY AND TIME

- Roman empire – Substitution ciphers (Caesar cipher)
  - Al Kindi (800 AD) – Frequency analysis
- 1400 AD – Voynich manuscript
  - Still undeciphered – might not be even cipher
- Early 20th century – great progress in both cipher construction and cryptanalysis
- 1917 Gilbert Vernam – one time pad
  - The only fully mathematically proven unbreakable cipher
- WW II – Enigma machine
- 1945 Claude E. Shannon – A mathematical theory of cryptography
- Modern cryptography – algorithm is public, only key is secret
CRYPTOGRAPHY AND TIME

- 1975 – DES by IBM, NIST, patched by NSA in 1977
  - Too small key size 56 bits – 1997 broken by brute-force attack by EFF
- 1976 – Diffie-Hellman
- 1977 – RSA
  - Currently still not broken but required key sizes grow
- 1994 – RC4
  - First problems found as early as in 1995
- Hash functions
  - 1990s - MD2,4,5 (Ronald Rivest), SHA1 (NSA)
    - All broken with less than brute force attacks
  - SHA2 (NSA), SHA3 (NIST competition) – unbroken
- 2025-2030? - Quantum computing – RSA, FF DH, EC broken
CRYPTOGRAPHY AND TIME

Encrypted protocols history


- SSLv2
- SSLv3
- TLSv1.0
- TLSv1.1
- TLSv1.2
- TLSv1.3
PERSPECTIVES
PERSPECTIVES

- Progress in design of new ciphers goes hand in hand with progress in cryptanalysis
- Algorithms and protocols will be broken and replaced
- Get accustomed with it
- Huge pool of legacy things to communicate with
CRYPTO POLICIES
WHAT IS IT?
CRYPTO POLICIES
What is it?

- A set of configuration files or configuration file snippets
- Centrally managed on the system
- Multiple pre-designed policy levels
- Core crypto components are covered
- Current Fedora releases, RHEL – 8
CRYPTO POLICIES
WHY DO WE NEED IT?
CRYPTO POLICIES

Why do we need it?

• Cryptography and secure protocols are widespread in the operating system

• Multiple cryptographic libraries provide the implementation

• Relatively fast changes x multiple things to configure

• Legacy devices using insecure algorithms or protocols
DEMO TIME!
DETAILS
DETAILS
Back ends

- OpenSSL
- GnuTLS
- NSS
- Java
- Kerberos 5
- Bind
- OpenSSH client
- OpenSSH server
- Libreswan
## DETAILS
### Policy levels

<table>
<thead>
<tr>
<th>Policy level</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LEGACY</td>
<td>Legacy devices interoperability, RC4, 3DES &gt;= 64bit security</td>
</tr>
<tr>
<td>DEFAULT</td>
<td>Reasonable but interoperable default &gt;= 80bit security</td>
</tr>
<tr>
<td>NEXT (Fedora only)</td>
<td>Equivalent of RHEL-8 DEFAULT</td>
</tr>
<tr>
<td></td>
<td>Removes TLS-1.0, 1.1, requires DH &gt;= 2048 bits</td>
</tr>
<tr>
<td></td>
<td>&gt;= 112bit security</td>
</tr>
<tr>
<td>FUTURE</td>
<td>Conservative level, no SHA1, 256 bit ciphers only &gt;= 128bit security</td>
</tr>
<tr>
<td>FIPS</td>
<td>Special policy for use by fips-mode-setup</td>
</tr>
<tr>
<td></td>
<td>FIPS approved/allowed algorithms only &gt;= 112bit security</td>
</tr>
</tbody>
</table>
DETAILS
Policy levels

- Use DEFAULT
- Use LEGACY only if you really need it
- Use FUTURE to test compatibility of your newly created/deployed application
DETAILS
Policy levels

• Even LEGACY level does not enable everything or make your system insecure

• Certain things are disabled on library level (SSLv2, 3)

• Custom levels currently possible

  • But back-end configs have to be manually created
DETAILS

Files

- `/etc/crypto-policies` – current configuration
- `/usr/share/crypto-policies/<level>/...` – predefined back-end configurations
- `/etc/crypto-policies/local.d/` – snippets to add to predefined back-end configurations
  - Currently used for adding p11-kit-proxy to NSS
CRYPTO POLICIES
WHAT IT ISN'T
CRYPTO POLICIES
What it isn't

- The crypto policies won't make your system magically secure
  - Still need to properly handle system updates
- No support for data at rest encryption
- It cannot magically configure things it does not know about
  - Unknown algorithms
  - Third party crypto implementations, ...
FUTURE
FUTURE

- Custom crypto policies
  - Building from policy definition files
  - Deployment
- Cooperation with other Linux OSes
- Support for more algorithms
- More back ends – Libssh, Go, ...
- Support for data at rest?
## RESOURCES

<table>
<thead>
<tr>
<th>Upstream project</th>
<th><a href="https://gitlab.com/redhat-crypto/fedora-crypto-policies">https://gitlab.com/redhat-crypto/fedora-crypto-policies</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>FIPS</td>
<td><a href="https://csrc.nist.gov/publications/detail/fips/140/2/final">https://csrc.nist.gov/publications/detail/fips/140/2/final</a></td>
</tr>
<tr>
<td>Manual page</td>
<td>man update-crypto-policies</td>
</tr>
<tr>
<td></td>
<td>man crypto-policies (next release)</td>
</tr>
</tbody>
</table>
TAKE AWAY

• System-wide crypto policies help with:
  • Maintaining your crypto usage up-to-date
  • Legacy compatibility
  • Preparation for future

• If something does not work:
  • update-crypto-policies --set LEGACY

• But otherwise please use DEFAULT
QUESTIONS?
THANK YOU

plus.google.com/+RedHat
linkedin.com/company/red-hat
youtube.com/user/RedHatVideos
facebook.com/redhatinc
twitter.com/RedHatNews