LSS 2016: linux-integrity subsystem status

Mimi Zohar

Linux Integrity Subsystem Status Update

- Continuing to close measurement/appraisal gaps
- Keyring changes
- Code signing: distro mirroring with file signatures
- Keys: pre-built kernel images
- Finer grain signature verification
- TPM 2.0 impact
- Summary

Continuing to close measurement/appraisal gaps

- (*Upstreamed*) Replaced existing hooks with a single set of pre & post security_kernel_read_file hooks

- (*Upstreamed*) Signed IMA policy
- (*New*) kexec support:
  - using new post security_kernel_read_file hooks for measuring/appraising the kernel image & initramfs
  - *(Upstreamed)* New PCR policy option (Eric Richter)
  - device tree support for preserving the measurement list (Thiago)
  - serializing/restoring the measurement list across kexec

- *(New)* other types of measurements, and maybe appraisals (eg. boot command line, userspace measurements, BPF?)

- CPIO/initramfs xattr support (with help from Victor Kamensky)

Closing measurement/appraisal gaps, and yet ...

- Ability to close a class of measurement/appraisal gaps with
  the new pre and post security_kernel_read_file hooks,
  but the hooks still need to be used (eg. LSM policies, ?).

- Continuing to close measurement/appraisal gaps,
  but new gaps are still being upstreamed.

  – Previously defined new syscalls with file descriptors
  – How different is unprivileged BPF than
    proprietary firmware, kernel modules?

Keyring changes

- IMA keyrings: MOK, blacklist (Petko Manolov, Mark D. Baushke)

- Generalized “trusted” key concept for restricting asymmetric keys: (David Howells)

  restrict_link_by_signature(),

  restrict_link_by_builtin_trusted(),
  restrict_link_by_builtin_or_secondary_trusted()

- Replaced system_keyring with builtin_trusted_keys (David Howells)
- Replaced MOK with global secondary_trusted_keys (David Howells)

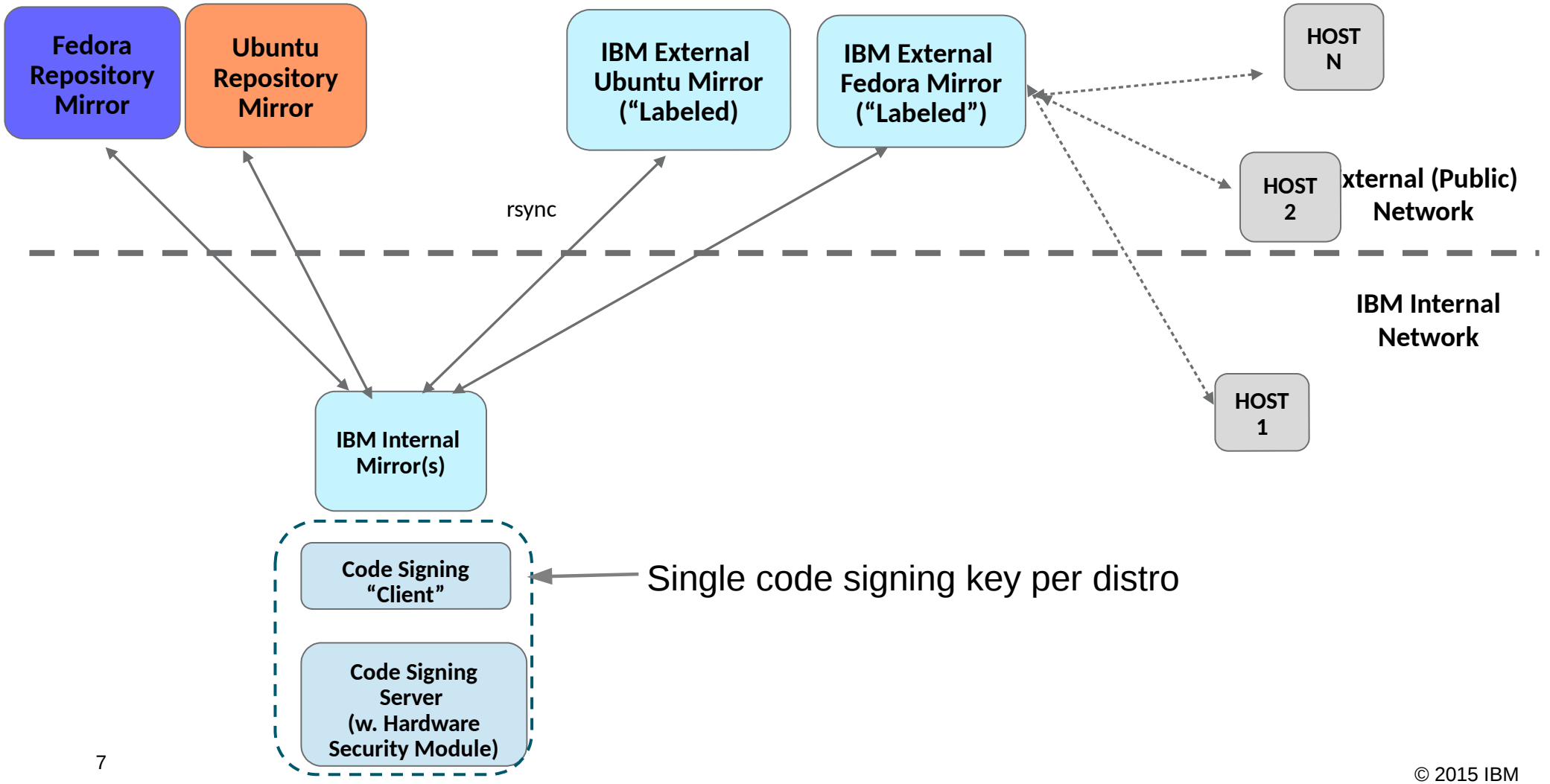
- Still need global revocation/backlist keyring

Keyring changes: userspace

- Similarly, generalize concept of restricting asymmetric keys for userspace (Mat Martineau, Intel)
  - Defines a new key/keyring root of trust for a userspace keyring
  - Extend secure boot signature chain of trust to the kernel and from the kernel to userspace

Code Signing: distro mirroring with signed files

Stefan Berger, Mehmet Kayaalp, Dimitrios Pendarkis
(IBM Research)



Keys: pre-built kernel images

- builtin_trusted_keyring only contains the kernel module key
- “distro” mirroring adds another key for all other signed code
- Including other certificates, in a pre-built kernel image, post build (Mehmet Kayaalp)
  - Requires reserving memory in the image
  - Inserting key into the image
  - Re-signing kernel image

Finer grain signature verification?

- Signatures not only verified by a key on a trusted keyring, but by a particular key

  .
- How granular will that key be: distro, package, package version or as suggested at the file pathname level?

- Need for a common policy definition method (please)

- Key management support for revocation/blacklists

TPM 2.0 impact (Nayna Jain, Ken Goldman)

- Algorithm agile - multiple TPM banks, one per algorithm
- Impacts the IMA measurement list
  - Larger digests
  - Multiple digests
- “Tainting” or extending other banks with padded/truncated hash?

Summary

- Ability to close a class of measurement/appraisal gaps with pre and post security_kernel_read_file hooks.

- Continuing to close measurement/appraisal gaps, but new gaps are still being upstreamed.

- Distro mirroring: including file signatures in packages is doable

- Still plenty to do:
  - What level of file signature key granularity? (revocation/blacklists)
  - Simplify CPIO/initramfs xattr support
  - TPM 2.0 impact
  - Namespacing IMA
  - Directory protection (Dimitrios Kasatkin)

Questions?

Thank you!