systemd & TPM2 in 2024

Linux Security Summit 2024, Vienna
Lennart Poettering

Update

- What’s new since May 2023
- Follow-up for talk at LSS NA 2023 in Vanvouver, Canada

Goals Recap

- Catch up with other Oses
- Measured Boot as Default
- Disk Encryption locked to TPM/Measured Boot by default
- Prepare Ground for Useful and Usable Remote Attestation

Quick Overview Status Quo Ante

- systemd-cryptsetup → unlock LUKS via TPM2
- systemd-cryptenroll → enroll TPM2 in LUKS
- systemd-pcrextend → measure fs identity, machine identiy, boot phases,… during boot, runtime, shutdown
- systemd-stub →EFI „stub“ that is glued in front of UKIs and measures the kernel components it is booting into
- systemd-measure →predict the measurements systemd-stub will make given a UKI, offline. Sign them, for use in LUKS TPM2 policy.

Status Quo Ante #2

- Ukify → glue systemd-stub, kernel, initrd, together to turn it into a UKI, then add systemd-measure signature into it, and SecureBoot-sign it.
- systemd-repart: automatically create partitions, file systems, and encrypt them with LUKS agains TPM2, intended use is at first boot.

What’s new? #1

- **systemd-pcrlock**: dynamic, locally managed PCR policies, which can be stored in a local TPM2 NV index, and directly referenced from TPM2 policies via AuthorizePolicyNV
- As opposed to signed PCR policies (which we previously supported) which are UKI (i.e. OS) vendor managed

systemd-pcrlock

- systemd-pcrlock can cover inherently local boot components (i.e. firmware of system + extension cards, but also local configuration, ...) which cannot reasonably be covered by OS vendor
- Disk Encryption policies can now lock to combination of „systemd-pcrlock“ policies and signed PCR policies.

Difficulties

- Couldn’t figure out a way to combined PolicyAuthorizeNV + PolicyAuthorize
- Solution: key sharding. One half unlocked via „systemd-pcrlock“, the other via signed PCR policy.
- Thus: local policy and vendor policy on equal footing

systemd-pcrlock is extensible

- systemd-pcrlock via drop-ins, covering various components of the boot, each with one or more variants.
- It’s careful, trying to not generate invalid policies.
- BTW, systemd-pcrlock already provides protection against software rollbacks

What’s new? #2

- systemd now manages its own **measurement log** in /run/.
- Uses JSON TCG CEL (almost, some trivial omissions, for reasons)
- Basis for systemd-pcrlock’s policy logic

What’s new? #3

- TCG offered to assign **static NV index range** to Linux as a whole. Currently in process to assign it to UAPI group.
- Delegate some from that range to systemd.
- Use for additional „fake PCRs“ (now called „NvPCRs“ in systemd), with the same semantics and guarantees.

NvPCR uses

- Infra PR pending (but needs rework)
- Measuring SMBIOS identity (also pending)
- Measure systemd-sysext, systemd-confext, portable services, systemd-nspawn containers on invocation (not done)
- NvPCRs means PCRs that aren’t quite „expensive“ anymore

What’s new? #4

- Well-defined **tpm2.target** unit, for cases where TPM2 access is not available unconditionally, i.e. to cover for .ko module loaded late, or for TPMs implemented in some local enclave/TEE or similar, which need userspace components to work.
- systemd-tpm2-generator tries to auto-detect if firmware recognized a TPM device, and inserts the target at the right places

What’s new? #5

- Multi-Profile UKIs
- Allow multiple combinations of initrd, cmdline, DeviceTree, … to be provided by a single UKI.

What’s Still Left

- Reasonable way to handle kexec & soft-reboot regarding measurements & sealing to PCRs
- Rotation of the measurement log
- Immutable disk encryption locked to a specific system’s PCR state, for use in systemd-confext for secure configuration deployment.

The End