IBM T J Watson Research Center

Using IMA for Integrity Measurement and Attestation

David Safford, Mimi Zohar, Reiner Sailer

Integrity Measurement Architecture (IMA)

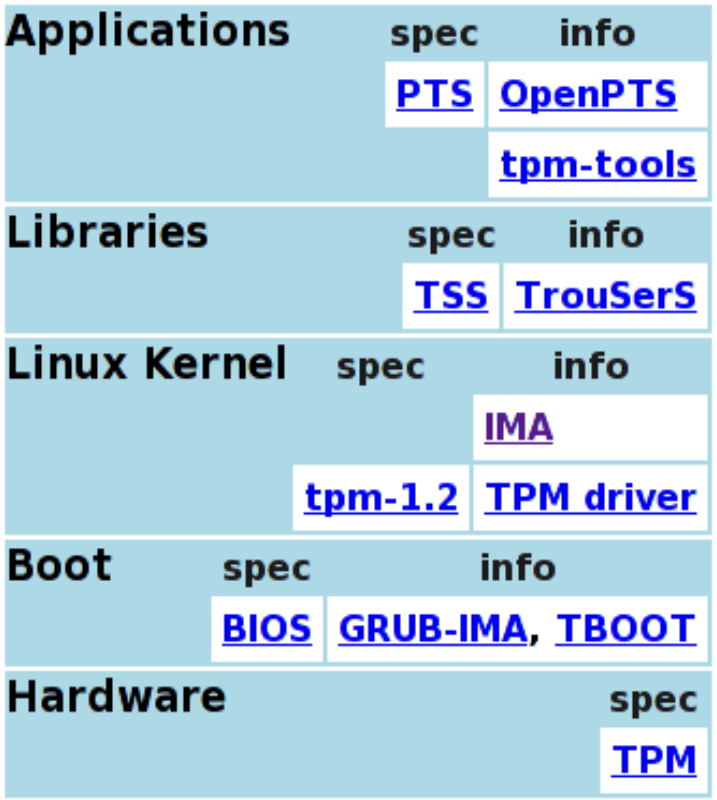
- What is IMA?
  - New kernel feature as of 2.6.30
- Enabling IMA
- Using IMA
  - Standalone or with TPM
    - health check - PTS
    - Network admission - 802.1x-TNC-PTS
- Future Work
  - EVM

http://linux-ima.sourceforge.net/

Trusted Computing: architecture & opensource components

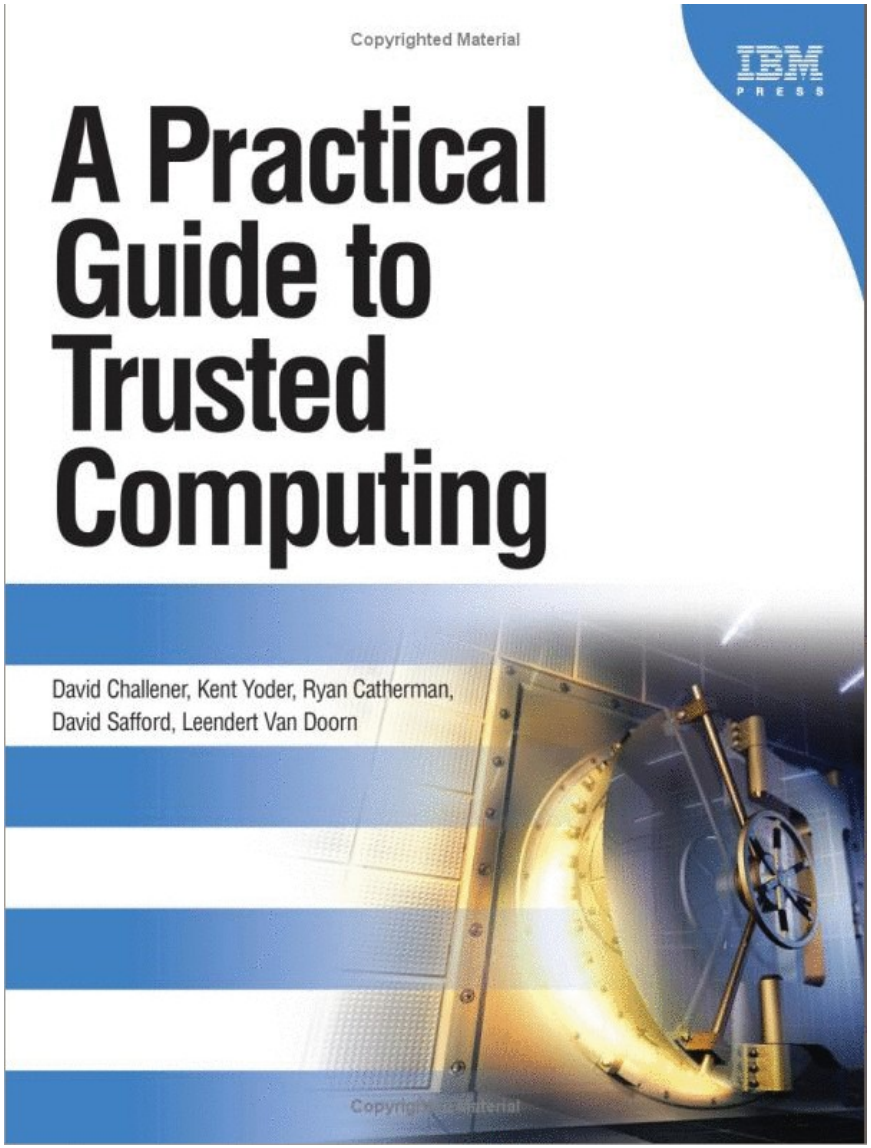
IMA maintains a list of file hash values and, if the system has a TPM chip, maintains an aggregate integrity value over this list inside the TPM hardware. The integrity measurement list can be read through a securityfs file, typically mounted at /sys/kernel/security/ima. The aggregate integrity value, normally in Platform Configuration Register(PCR) 10, can be signed using the TPM quote, so that the measurement list can be cryptographically verified. Together the measurement list and the signed aggregate integrity value can be used to attest to a system's runtime integrity.

IMA's maintenance of a TPM hardware anchored file measurement list is fundamental to TCG's Platform Trust Services(PTS) and, not shown here, Trusted Network Connect(TNC) standards.



A Blatant Plug

- Programming
  - BIOS
  - Device Driver
  - TPM
  - TSS
- Applications
  - Trusted Boot
  - Key Management
  - Authentication
  - Attestation



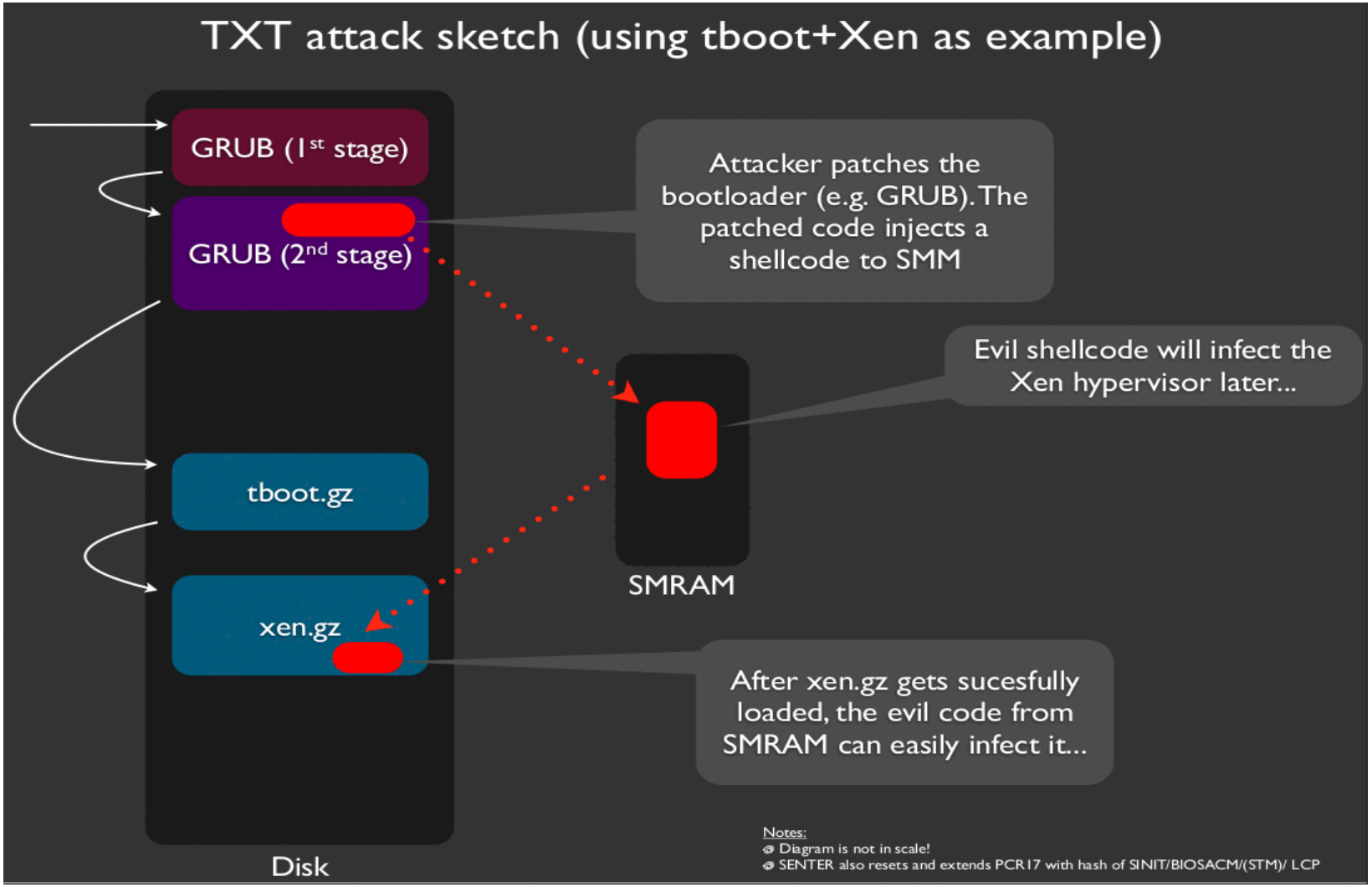
Trusted Platform Module (TPM)

- RSA crypto
  - key generation, signature, encrypt, decrypt
- Secure storage
  - private keys
  - master keys (eg loopback)
- Integrity measurement
  - Platform Configuration Registers (PCR)
  - compromise detection
  - Tie key use to uncompromised environment
- Attestation
  - host based integrity/membership reporting
  - (RSA 2004 Demo)

TPM as a Root of Trust

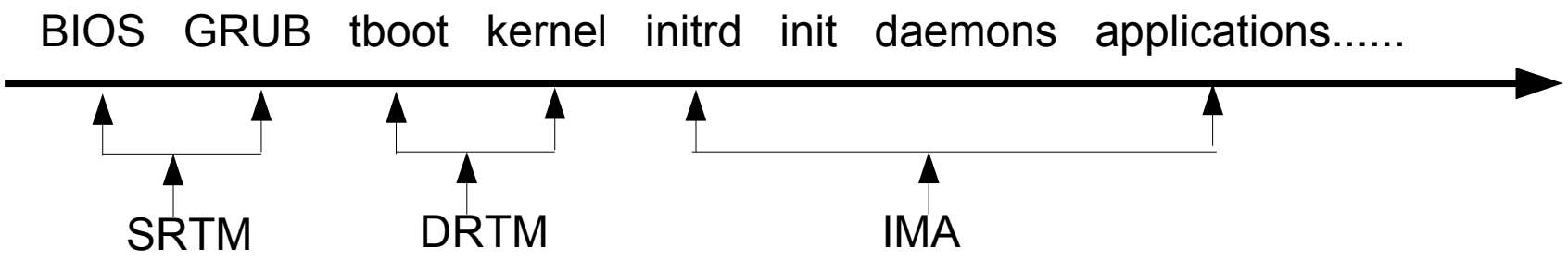
- Static Root of Trust (SRTM)
  - Immutable BIOS measures mutable BIOS
  - Each step thereafter measures the next stage
- Dynamic Root of Trust (DRTM)
  - Atomic measure/load/execute bootstrap
  - Not dependent on BIOS
  - But: Rutkowska, “Attacking Intel's Trusted Execution Technology” Blackhat 2009

SMM Attack on DRTM (Rutowska, Blackhat 2009)



Addressing the generalized problem

- “Fixing SMM” addresses just this set of attacks
- If you run someone else's program on your computer, it's no longer your computer
- If you don't know what's running on your computer, you cannot know if it is still your computer
- This includes ALL programs, at ALL levels



Integrity Measurement Architecture (IMA)

- Trusted Computing Group Trust Architecture
  - Chain of trust – measure files before accessed/executed
  - Store measurements in kernel list
  - Extend measurements into TPM/vTPM PCR
  - Attest all measurements to third party, signed by TPM/vTPM
  - Malware cannot take measurements back from TPM/vTPM
- IMA is linux kernel module which implements this model
  - Policy based for which files to measure
  - High performance with measurement caching

Using IMA – the basics

- Config/install the kernel
  - Integrity Measurement Architecture(IMA) (IMA) [N/y/?] (NEW)
  - (IMA and TPM drivers must be compiled in, not modules)
- Add “ima_tcb=1” to kernel boot
- Mount securityfs
- If desired load tailored policy
  - /sys/kernel/security/ima/policy
- Read measurement list
  - /sys/kernel/security/ima/ascii_measurement_list

IMA Raw output

```
10 1609097048173d7c1659ff30713741b98020d0d8 ima a36e075c9dc23bd71886d55e0daee556143f8e7e boot_aggregate
10 3d82394c528268a62687c8b59b34c39137b4fb3c ima ff65625e34131617ef3ac5fead5cb285b6aa73b9 /init
10 231309f206f12296cff6c59c6e3bceb45dc285bb ima 602acdac8864b113d4546bf8f2d7b96c81607792 /init
10 9d20d222ae9a3a9c80b2a9f0b3c08a5786847ed6 ima ff4c31959c04f865c859d8df331a6bf6b6967cef ld-2.10.1.so
10 88a2bee776f4bd1305ce6ded08611fdecb5bf0db ima 44e727b6c99370d373d0acef95631e0950ef8c00 ld.so.cache
10 0c86c858b30abf3f2c3b0269a56204f642f3ef71 ima a0ed012c34fee03e9fecee8d4a7bcab1ff98f091 libnash.so.6.0.87
10 676a27e4112c9fb5677b7ef38cc488d87fca0846 ima bf094171a04b06d642c931d7c2ce0a707a659827 libbdevid.so.6.0.87
10 04c918d9ccb56ca92a118b72f8b2f60da0791887 ima 9355215ed19d72287c446d0f3b7b41dcaddbd254 libdevmapper.so.1.02
10 6585544486aee180aa950b1bb6132434cf0f71d6 ima caca8bbf8ff4957584d114c753b7c4f15936af7c libparted-1.8.so.8.0.0
```

After typical boot (Fedora 11), 1600 measurements.

Overhead at boot time, 10% (3-5 seconds).

Slight performance improvement at runtime (a lot prefetched).

IMA Measurement Policy

- Want to measure all files, but unacceptable performance
- Some measurement decisions are easy:
  - All executed files, #! scripts (bprm hook)
  - All files mmap'ed executable (mmap hook)
- Some read()'s are sensitive, but not all...
  - scripts, config files are sensitive
  - NOT – log files, LARGE files (KVM images...)

Need a measurement policy integrated with LSM, to take advantage of selinux subject, object, type labels

IMA Policy language

```
action: measure | dont_measure
condition:= base | lsm

     base:   [[func=] [mask=] [fsmagic=] [uid=]]

         func:= [BPRM_CHECK][FILE_MMAP][PATH_CHECK]
         mask:= [MAY_READ] [MAY_WRITE] [MAY_APPEND] [MAY_EXEC]
         fsmagic:= hex value (or NAME)
         uid:= decimal value

     lsm:    [[subj_user=] [subj_role=] [subj_type=]
          [obj_user=] [obj_role=] [obj_type=]]

         these are LSM specific

Omitted conditions match any, if no matching rule then dont_measure
```

IMA ima_tcb=1 default policy

```
dont_measure fsmagic=PROC_SUPER_MAGIC
dont_measure fsmagic=SYSFS_MAGIC
dont_measure fsmagic=DEBUGFS_MAGIC
dont_measure fsmagic=TMPFS_MAGIC
dont_measure fsmagic=SECURITYFS_MAGIC
dont_measure fsmagic=SELINUX_MAGIC
measure func=BPRM_CHECK
measure func=FILE_MMAP mask=MAY_EXEC
measure func=PATH_CHECK mask=MAY_READ uid=0
```

Example LSM Specific Measurement Policy

```
SELinux:
dont_measure obj_type=var_log_t
dont_measure obj_type=auditd_log_t

Smack:
measure subj_user=_ func=INODE_PERM mask=MAY_READ
```

IMA messages – when IMA **Can't** Measure a file

- The kernel prohibits writing and executing a file concurrently
  - Other files can be read and written concurrently
- "open_writers" - file already open for write, is opened for read
- "ToMToU" ("open_reader") – file already open for read is opened for write
- In these two cases, IMA cannot know what is actually read, and invalidates the measurement with all zeros
- Applications that do this have no idea what they are reading
  - Possibility of failure, cross domain exploit?
  - Gnome-pty-helper -> utmp, configure -> sh
- In cases so far, just tweak policy not to measure

IMA with TPM

- TPM device driver (TPMDD) – in kernel
- TPM library (TSS) – Trousers
  - TPM tools, utilities – included with Trousers
  - TPM_QUOTE can be used to sign measurement chain pcrs

Detecting and Isolating Compromised Systems

- Boot Time Integrity Measurement and Attestation
  - TPM based SRTM and DRTM
- Run Time Integrity Measurement and Attestation:
  - IMA – Integrity Measurement Architecture (2.6.30)
    - Measure all files before they are used
    - TPM based attestation of measurement list (PTS)
- Network Admission/Isolation Time
  - 802.1x-TNC-PTS standards compliant attestation for network admission
    - Trusted Network Connect (TNC)
    - Platform Trust Services (PTS)

Using Platform Trust Services (PTS)

- TCG XML standard for reporting integrity measurements
  - IMA measurement list and TPM_QUOTE
- How do you know “good” measurements?
  - Managed clients
    - Comparison to managed image
    - Reference Manifest vs current report (ie what’s changed?)
    - Comparison to reference database (nsrl.nist.gov)

Attestation 802.1x-TNC-PTS

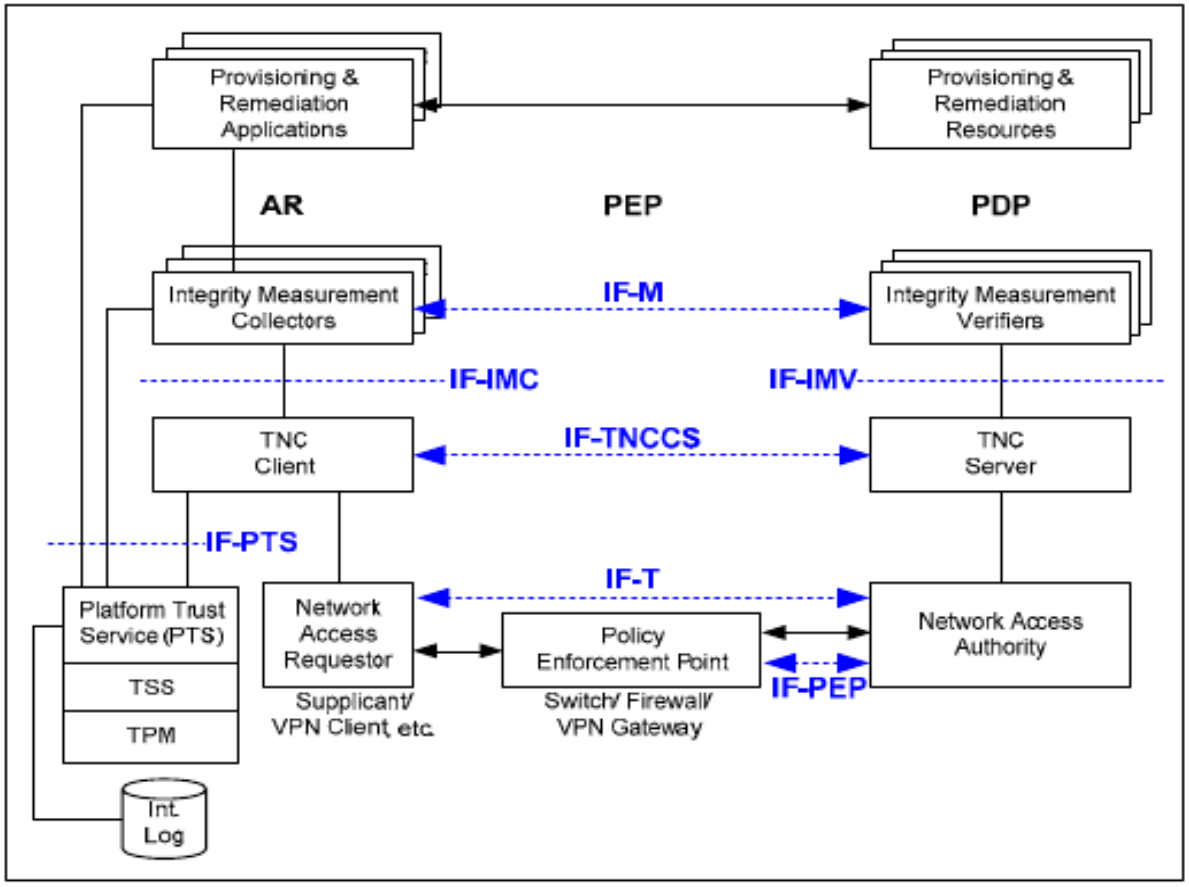
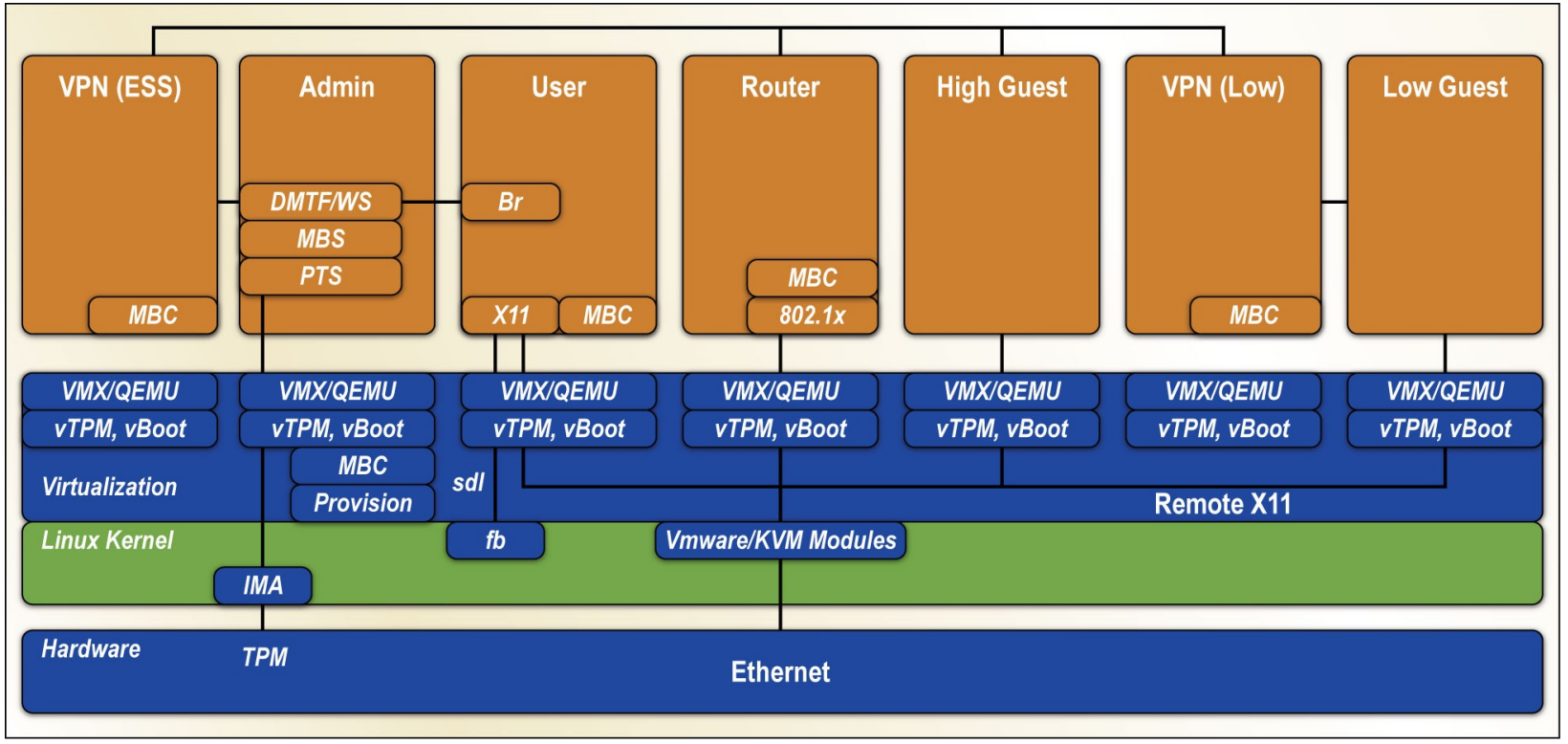


Figure 5: The TNC Architecture with the Trusted Platform Module (TPM)

802.1x-TNC-PTS in Virtualized Environments



09-0117-005

802.1x-TNC-PTS-IMA Resources

- OpenPTS (Seiji Munetoh, TRL)
  - http://sourceforge.jp/projects/openpts/
- 802.1x-TNC Client: wpa_supplicant
  - http://hostap.epitest.fi/wpa_supplicant/
- 802.1x-TNC Server: tnc@fhh
  - Fachhochschule Hannover
    (University of Applied Sciences and Arts)
  - http://trust.inform.fh-hannover.de/joomla/
- Libtnc
  - http://sourceforge.net/projects/libtnc

Future work – EVM – Local "Appraisal"

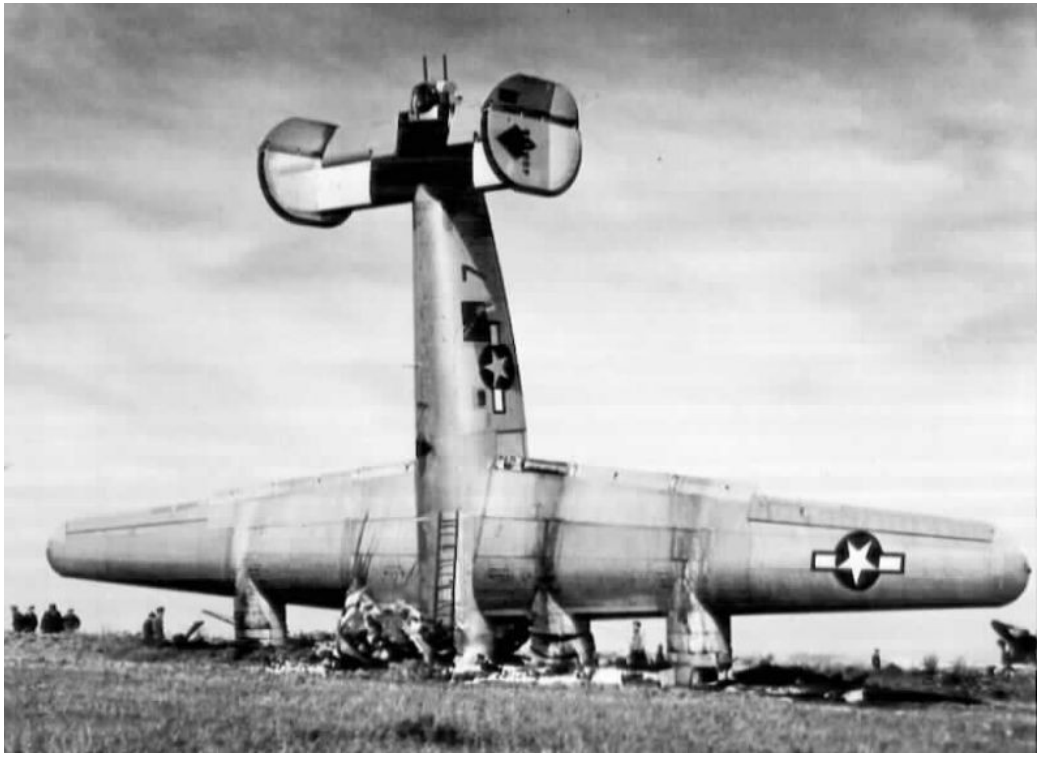
- Verification of a file's data and metadata (lsm labels)
- Original EVM
  - Xattr stored HMAC on data hash
  - Initial labeling slow, and directory level attacks
- Fixing Design
  - Scalability - policy extension – appraise/dont_appraise
  - HMAC or RSA signatures in xattr
  - Appraise directories – performance critical path

Summary

- IMA maintains trust chain in kernel
- Enabling IMA
- Using IMA
  - Standalone or with TPM
    - health check - PTS
    - Network admission - 802.1x-TNC-PTS
- Future Work
  - EVM

BACKUP

Introduction: State of Computer Security



"The sky isn't falling ... it fell a few years ago."
Roger Grimes, Infoworld Security Advisor, 2006

The modern threat

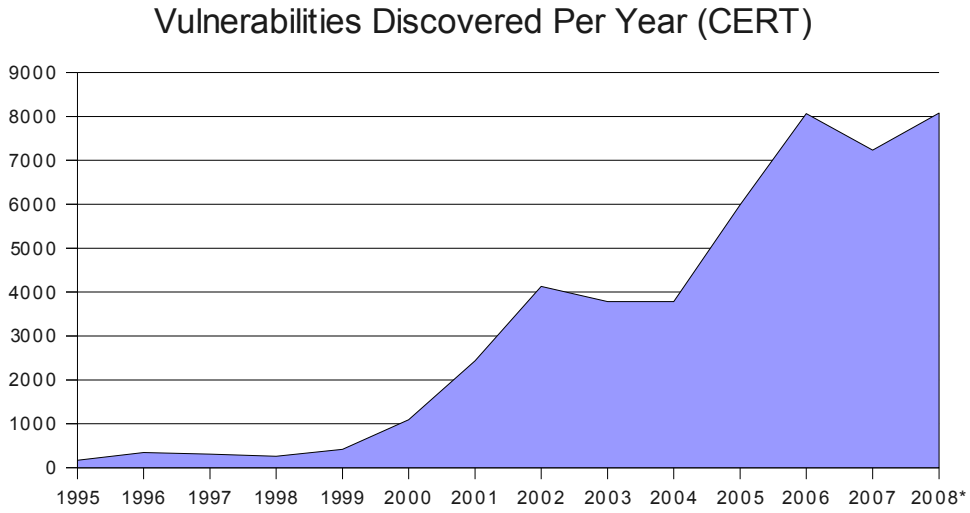
"Nation states, however, have the technical and operational capabilities to orchestrate the full range of adversarial cyber operations through a combination of such means as recruiting insiders, setting up front companies, establishing signals collections systems, implanting damaging hardware or software in communications networks and subverting telecommunications, cryptographic defenses and supply chains."

National Science and technology Council, "Federal Plan for Cyber Security and Information Assurance Research and Development", April 2006.

The Ease of Application Hacking

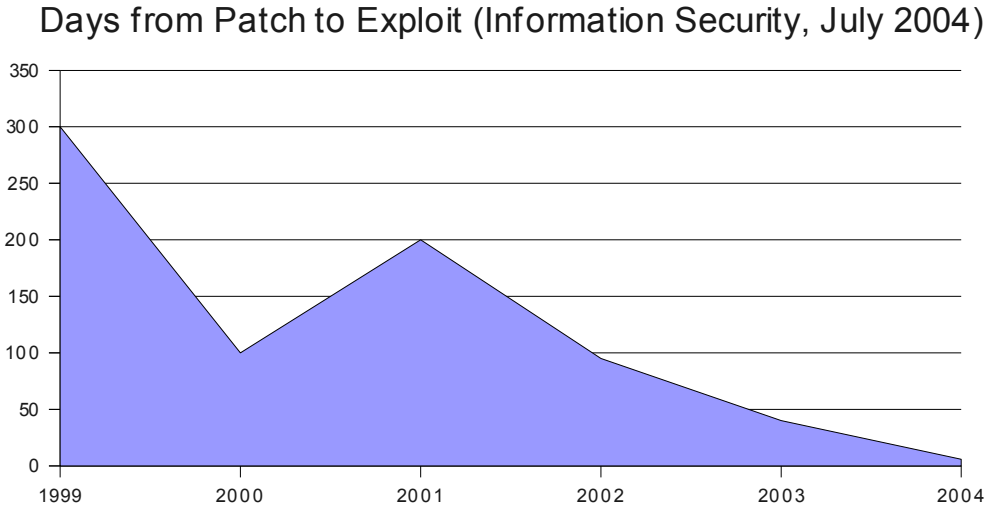
- Attacking Servers:
  - 97% web sites vulnerable to SQL injection or XSS. - IBM ISS
- Attacking Clients:
  - Chinese Hacking
    - Spear phishing with Word and PDF exploits
    - 1,295 (known) PC's in 103 Countries
    - High value targets
    - Remarkably simple, effective attacks

We've Lost the Software Vulnerability War



More and More Vulnerabilities (roughly 20 per day)

Less and Less Time to Patch (zero day exploits)



Secure Software is HARD

- So far, every software system has failed
  - Apollo Command Module Computer (16K words) failed every flight
- Studies shows at least 1 bug per K lines of code (LOC)
  - IBM internal study, 2000
  - Information Week Jan 21 2002, p23
  - Reasoning, Inc 2003
  - coverity.com 2008
- Linux and WinXP with Office each have > 200MLOC
  - 400K bugs would take 80 years @ 5000/year to fix
  - But we are writing roughly 50K new ones per year!
- Can model this as an infinite supply of security bugs.
  - Must design our systems to handle vulnerable software

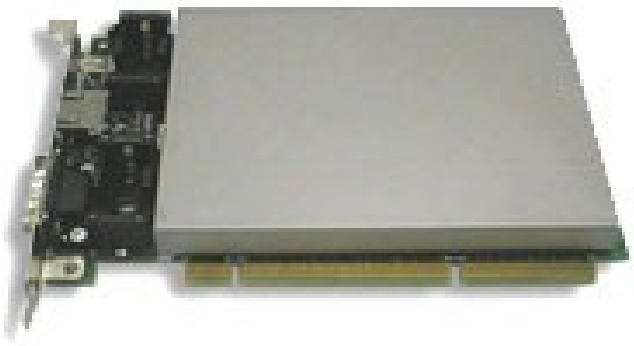
The Failure of Secrecy

- "Three may keep a secret, if two of them are dead."
  Benjamin Franklin, 1735
- Benjamin was a hopeless optimist.
  - Individuals seem delighted to give away their secrets.
  - Phishing/pharming
  - Gartner: $3.2B losses, 3.6M victims of phishing in 2007

- "One may keep a secret, if he doesn't know what it is."
  Dave Safford, 2004 - TPM

Hardware Based Security

IBM PCI-X Cryptographic Coprocessor



- **Announced in September, 2003**
- **Greatly improved performance**
- **PCI-X and network interface**
- **Same physical / logical security feature set as 4758**
- **Received FIPS 140-2 Level 4 validation**
- **Support for IBM zSeries (mainframes) today**

Secure Hardware

Integrity Aware Parallelizable Mode (IAPM)

- Originally developed for network communications
  - With “almost free” integrity
- Use “whitening” with pairwise independence
  - Builds “location sensitivity” into ciphertext
- Processed in parallel and/or pipelined engines
  - Both encryption & decryption
- Submitted to NIST for evaluation as a block cipher mode