Lecture notes for data remanence

Threat

* Security-critical data stored in memory
* OS mediates access to memory
* Attacker uses physical access to bypass OS mediation
* Basic assumption that DRAM loses values immediately when unpowered is incorrect
* Laptops w/ encrypted drives particularly good targets

Cold boot threat

* Attacker reboots into very small kernel
* Then reads out all memory values
* Data survives the brief power cut at reboot
* [Unsafe] reboot via power cut prevents OS from scrubbing memory at shutdown

Imaging memory

* After rebooting into small kernel, dump memory
* Simply read memory, write to storage
* Tools easily deployable, eg hidden on an iPod

Decay rate

* DRAM technology requires refresh to keep value
* In absence of refresh, values decay to ground states
* Decay rate varies with temperature

Measured decay

* Operating temperature (25c to 45c): seconds
  + Fastest 2.5 sec
  + Slowest 35 sec
* -50c: 1% decay after 10 mins
  + Cooled with “canned air” aerosol
* -196c: 0.17% decay after 60 mins
  + Cooled with liquid N

Key extraction

* Identify AES, DES, & RSA keys
* Algorithms recover from 10% bit error in seconds

Key identification

* Keys look like random bits, how can we distinguish them from actual random bits?
* Search for blocks of memory matching expected key scheduling algorithm

Cryptographic file system breaks

* BitLocker (Microsoft)
* FileValut (OS X)
* TrueCrypt (open source)
* Dm-crypt (Linux)
* Loop-AES (Linux)

Countermeasures

* Challenge: crypto keys in use need to be stored somewhere
* Scrub memory during mobo POST
* Prevent network boot, prevent removable media boot
* Destroy keys at suspend, require user to reenter secret to recreate keys
* Algorithmic implementation changes to key scheduling
* Change the physics of MOSFET (metal oxide semiconductor field-effect transistor)
* Encrypt in disk controller, not in main memory
* Encrypt in TPM