**Side-channel Attacks**

Side-channel Attacks: Information leakage through electromagnetic signals, shared memory/registers/files between processes, CPU usage metrics

- Keyboard Electromagnetic emanation [1], Keystroke acoustic [14]

- Inter- keystroke timing information in Linux (the stack pointer ESP of a process profiling) [2]

- Estimation of the cache usage, traffic load, and keystroke timing between co-resident VMs [3]

- RSA secret keys extraction from OpenSSL [4]

- Extracting different SSH keystroke-combinations from an interactive remote shell service and slight timing characteristics [5]

- Voice-over-IP using variable-bit-rate encoding scheme [6]

- *Slingbox Pro*, a device for encrypted video- streaming, allows the attacker to determine the title of the movie being played [7]

Side-channel leaks from encrypted traffic based on fingerprinting

- Against SSL/TLS [8-11]

- Against WPA and IPSec [12]

- Side-Channel Leaks in Web Applications: A Reality Today, a Challenge Tomorrow [13]

- Memento: Learning Secrets from Process Footprints [14]

Smartphone

- Information leak using sound [16]

Etc

- LCD reflection [15]

Multimedia Decoding

- Variable MPEG decoding CPU statistics: Inverse Discrete Cosine Transformation (CPU-intensive) [17]

- Motion Compensation and IDCT are the most expensive modules [18]

[1] Martin Vuagnoux and Sylvain Pasini. Compromising Electromagnetic Emanations of Wired and Wireless Keyboards. USENIX Security Symposium, 2009.

[2] Kehuan Zhang and XiaoFeng Wang. Peeping Tom in the Neighborhood: Keystroke Eavesdropping on Multi-User Systems. USENIX Security Symposium, 2009

[3] T. Ristenpart, E. Tromer, H. Shacham, and S. Savage. “Hey, You, Get Off of My Cloud! Exploring Information Leakage in Third-Party Compute Clouds.” ACM CCS 2009.

[4] D. Brumley and D. Boneh. "Remote timing attacks are practical," the 12th Usenix Security Symposium, 2003

[5] Dawn Song, David Wagner, and Xuqing Tian. "Timing Analysis of Keystrokes and SSH Timing Attacks," 10th USENIX Security Symposium, 2001

[6] Charles Wright, Lucas Ballard, Scott Coulls, Fabian Monrose, and Gerald Masson. "Spot me if you can: recovering spoken phrases in encrypted VoIP conversations," in IEEE Symposium on Security and Privacy, May, 2008.

[7] T. S. Saponas, J. Lester, C. Hartung, S. Agarwal, and T. Kohno. "Devices That Tell On You: Privacy Trends in Consumer Ubiquitous Computing," Usenix Security, 2007.

[8] David Wagner and Bruce Schneier. Analysis of the ssl 3.0 protocol. The Second UNIX Workshop on Electronic Commerce, pages 29–40. USENIX Association, 1996.

[9] Heyning Cheng, Heyning Cheng, and Ron Avnur. Traffic analysis of ssl encrypted web browsing, 1998.

[10] George Danezis: Traffic Analysis of the HTTP Protocol over TLS. http://research.microsoft.com/en-us/um/people/gdane/papers /TLSanon.pdf

[11] Qixiang Sun, Daniel R. Simon, Yi-Min Wang, Wilf Russell, Venkata Padmanabhan, and Lili Qiu, "Statistical Identification of Encrypted Web Browsing Traffic," in IEEE Sym. on Security & Privacy 2002.

[12] George Bissias, Marc Liberatore, David Jensen, and Brian Neil Levine. "Privacy Vulnerabilities in Encrypted HTTP Streams." Privacy Enhancing Technologies Workshop (PET), May 2005.

[13] S. Chen, R. Wang, X. Wang, and K. Zhang. Side- channel leaks in Web applications: A reality today, a challenge tomorrow. In S&P, 2010.

[14] D. Asonov and R. Agrawal. Keyboard acoustic emanations. In S&P, 2004.

[15] M. Backes, M. Du ̈rmuth, and D. Unruh. Compromising reflections - or - how to read LCD monitors around the corner. In S&P, 2008.

[16] R. Schlegel, K. Zhang, X. Zhou, M. Intwala, A. Kapadia, and X.Wang. Soundcomber: A stealthy and context-aware sound trojan for smartphones. In NDSS, 2011.

[17] Kihwan Choi, Karthik Dantu, Wei-Chung Cheng, and Massoud Pedram. Frame-Based Dynamic Voltage and Frequency Scaling for a MPEG Decoder. ICCAD, 2002.

[18] Matthew J. Holliman , Eric Q. Li , Yen-kuang Chen. MPEG Decoding Workload Characterization. Proceedings of Workshop on Computer Architecture Evaluation Using Commercial Workloads, 2003. (CAECW)