Project proposal

Real-time signal-processing platform for compromising video emanations

Markus G. Kuhn – 2005-04

The facilities currently available in our hardware-security laboratory at the University of Cambridge Computer Laboratory for the demonstration of compromising video signals include:

- wide-band Tempest receiver (Dynamic Sciences R-1250, 100 Hz to 1 GHz carrier frequency, 50 Hz to 200 MHz bandwidth, AM/FM/BFO demodulation)
- logarithmic-periodic antenna (200 MHz to 1 GHz, tripod)
- digital storage oscilloscope (Tektronix TDS7054, 500 MHz input bandwidth, linked to a Pentium II PC)
- two programmable function generators (TTi TGA1230 and TTi TG1010, 0–10 MHz)

Back in 2001, I set up a simple demonstration experiment for visualizing, in real-time, compromising emanations from nearby video displays. I programmed a pair of digital function generators to produce video synchronization signals with frequencies that match those of the eavesdropped device. To obtain a stable image, it is necessary to adjust these frequencies continuously to within no more than 1 part-per-million error (i.e., 6 digits accuracy), thereby compensating for temperature-dependent frequency drift. This required the use of a fairly complicated configuration of two cascaded function generators, which are configured for a particular video standard by a computer software I developed [1, Appendix B.2]. The reconstructed clock frequencies from these generators are fed into to a standard VGA computer monitor (multisync CRT). The analog output of our wide-band receiver is then connected, via a simple impedance-matching adapter, to one of the monitor's RGB input lines. Using this setup, I can visualize to visitors, in real-time, compromising emanations from video displays, at least as long as the signal-to-noise ratio is not substantially below 0 dB.

Using the available digital storage oscilloscope, I can also perform measurements in which the periodic nature of the eavesdropped signal is exploited for significant noise reduction, using periodic averaging.

Problem

Unfortunately, the available oscilloscope was not designed for fast real-time averaging of recordings as long as a full video frame. This leads to a pause of several seconds between individual frame acquisitions. During this time, the data must be copied over a relatively slow interface out of the oscilloscope's acquisition memory. As a result, averaging merely a few hundred frames with this oscilloscope takes already well over 5 minutes. During such long measuring times, the temperature drift of the oscillators involved causes a substantial loss of synchronization, and accurate averaging is only practical when the oscilloscope is triggered directly from the eavesdropped video card. In addition, the resulting averaged data must finally be transferred from the oscilloscope into a PC, where a program turns it into a raster image that can be viewed on a monitor. This step adds another

several minutes of processing time. The entire procedure needs to be repeated until all the parameters have been adjusted well enough for an acceptable image quality, a process that might take several hours. While this approach is useful for analyzing the nature of compromising emanations in a laboratory setting, it is far from a practical, realistic and impressive demonstration of a real-world eavesdropping scenario.

The presently available real-time demonstration setup was improvised from donated equipment without any funding. It has a number of serious technical problems:

- The video signal supplied to the CRT is not blanked near the horizontal sync pulse. As a result, the clamping circuit in the CRT monitor distorts the image quite noticeably. This becomes visible as distracting dark horizontal stripes on the eavesdropping monitor, especially with strong background signals.
- The only exploitation of the periodic nature of the signal is made through the fact that the human eye is quite good at ignoring high-frequency background noise in the displayed image.
- The function generators need to be fed with a very complex programming sequence to produce a suitable video signal. This is only practical by connecting them to a PC, from where a script uploads their configuration settings.
- Changing the video parameters of the sync signals produced by the function generators takes more than five seconds, to the slow serial port interface over which these devices have to be controlled. This makes it very tedious to guess and try the video parameters of an eavesdropped device, as it is not possible to scroll through parameters, such as the line/frame frequency ratio, in real time.
- For bandwidths above 20 MHz, the input of the CRT used is not matched properly to the voltage range provided by the receiver, therefore the image remains for many interesting signals far too dark.
- The synchronization frequencies supported by the CRT do not cover the full range of frequencies of interest for working with repetitive compromising emanations (e.g., line frequencies used by printers).

A more impressive and adequate demonstration of the risk of compromising emanations will require additional, purpose-build equipment. There are some specialized function generators on the market that reconstruct sync signals with the required frequency accuracy and take care of blanking the video signal:

- RAS-515A Raster analysis system. Martin L. Kaiser Inc. Cockeysville, MD 21030, USA. http://www.martykaiser.com/ras515a.htm
- RG1000A, Raster Generator, Dynamic Sciences International Inc., Woodland Hills, CA, USA. http://www.dynamic-sciences.com/rg1000a.html

Both these devices only generate synchronization or deflection signals to raster the output of a receiver on a monitor or an oscilloscope in XY mode. They provide no digital memory to perform periodic averaging, or any other processing steps, on the video signal.

A family of more sophisticated commercial devices, possibly equivalent to the one proposed below, is sold by SystemWare Inc., California (http://www.sysware.com/signal.htm), in particular the

"(Enhanced) Digital Raster Library" and "Frame Control Library" that are offered as part of their "ScenarioFlex" signal intelligence and analysis hardware/software package.

Proposal

The most practical approach to solving all of the problems list above,seems to me to design a high-speed digital video-signal processing system that sits between the analog wide-band receiver and the video monitor. Advances in field-programmable gate array (FPGA) technology and analog-to-digital conversion (ADC) technology should make it feasible today to build a device with all the required functionality from standard components. In particular, several available FPGA evaluation boards featuring the Altera Stratix II or the Xilinx Virtex-4 FX FPGA with on-board RAM, ADC, DAC, VGA or DVI output and clock generators should make it possible to program almost the entire required circuit. With a suitably chosen off-the-shelf development board, the only required additional hardware may be an add-on daughter-board with faster ADC, as well as analogue matching amplifiers, that may have to be designed. One example of a suitable evaluation board, which I had so far a closer look at, is the

- Altera Stratix II EP2S60 DSP Development Board
  http://www.altera.com/products/devkits/altera/kit-dsp-2S60.html

which features 12-bit 125 MHz ADCs, a VGA graphics output, and 32 megabytes on-board DRAM. Several other boards would equally be suited.

The periodic averaging functionality can be implemented in two ways. One major design choice is between:

(a) Identical input and output sample frequencies. In this option, the entire circuit is entirely driven by a clock frequency that is adjusted via a cascade of phase-locked-loop (PLL) frequency synthesizers to match exactly a multiple of the frame rate of the eavesdropped device. For each cycle, a new sample voltage $s(x,y)$ is read from the ADC. This value is then accumulated in a pixel buffer $b(x,y)$, using the exponential-decay filter $b_{\text{new}}(x,y) = \kappa \cdot b_{\text{old}}(x,y) + (1 - \kappa) \cdot s(x,y)$, where $0 < \kappa < 1$ is an adjustable filter parameter [1, p. 97]. During the same cycle, the value $b(x,y)$ is sent to the VGA output for display, suitably blanked and scaled, along with the appropriately generated sync signals.

Advantages:

- This is by far the simplest design

Disadvantages:

- The clock frequency of the DSP board will be a multiple of the frame rate, therefore the electromagnetic interference generated by the board will not be eliminated by the averaging process. The board would have to be electromagnetically shielded very carefully, and even then there may remain noticeable cross-talk between the clock signals and the resulting video signal.
- The averaging periods will be restricted to the frame rates of which the output monitor is capable. This would prevent the device from being used to demonstrate periodic

emanations from devices other than video display systems (e.g., printers).

(b) Independent input and output sampling frequencies. In this option, the entire circuit is driven at a fixed clock frequency. A digital resampling unit is implemented, that uses linear interpolation to convert the sampling frequency of the stream of input pixels from its originally fixed frequency to an adjustable frequency that can match and track the pixel clock of the eavesdropped device.

Advantages:

- No clock signal in the averaging board will be phase-aligned with the pixel clock of the targeted device, therefore the electromagnetic interference can be eliminated by the periodic-averaging process.
- Depending on the exact implementation of the resampling unit, it may be possible to cover repetition frequencies well beyond the line and frame rates of the connected VGA display.

Disadvantages:

- This approach requires a considerably more complex design. A linear-interpolation unit with several multipliers has to be implemented. High-precision fixed-point arithmetic and registers have to be implemented to track the screen-position matching the input video signal. Reading and updating of the video RAM will be at independent locations, which may double the required RAM bandwidth.

While an initial prototype may be based on option (a), it would be highly desirable to attempt also the implementation of option (b). Ideally, the completed averaging board should have the following characteristics:

- Input video frequencies adjustable via line rate, frame rate, lines/frame ratio, preset VESA modes
- Generation of a VGA signal (and/or DVI, if supported by board), adequately blanked, with synch signals
- Support of several VGA output video modes, including 1600×1200 (60 Hz, 162 MHz), 1280×1024 (60 Hz, 108 MHz), 1024×768 (60 Hz, 65 MHz), and a mode for tracking the frequencies of the target
- Generation of a simple on-screen text display, for adjustment of parameters with four cursor keys or a mouse (via RS-232)
- Numeric brightness and contrast adjustments
- Automatic frequency tracking, for example using a cross-correlation of an individual line across frames, or of averages of lines; the feasibility of this will depend largely on the power of the NIOS CPU that will have to be loaded into the FPGA to control the configuration
- Support of remote-control and image download via 100 Mbit/s Ethernet
- A printer mode, which averages a small number of lines into a single output line and overwrites frames instead of accumulating them
- A fast spectrum-analysis mode for producing a “waterfall” diagram

While an early primitive prototype of merely option (a) without any CPU control and on-screen display can probably developed within about a month by a single developer (full-time), the full functionality outlined above will certainly take several person-months to develop. I myself can contribute to the initial stages of this project a substantial amount over the summer (starting mid July 2005), and I hope that one of my students can contribute some of the features that would not be part of the initial version.

Cost

Without having received any quotations yet, here is an early estimate of the equipment cost:

A single FPGA development board, of the type listed above, I expect to cost in the order of £2000. To allow a student to contribute to the project, and also to have a backup, should something break, the purchase of a second board would be crucial. One of my colleagues has an educational license for the Atmel Quartus II development software that is necessary to program the FPGA. If the project is performed purely as an academic research project, I may be able to use that, otherwise it may be necessary to license this software as well. In addition, to run effectively, I am being advised by my colleagues that this software will require a Windows PC with at least a gigabyte of RAM, which we do not have available. Another £1500 should be budgeted for that (including a DVI display), plus £2000 for materials and consumables needed to build a signal matching and ADC daughter board.

So without any labour cost, budgeted purely as an academic research project implemented by volunteers, the minimal funding necessary would be in the range of £7500. This budget does not yet include a commercial license of the ECAD software, should it be necessary.

For a faster development, or anything that results in intellectual property for the sponsor, I would have to hire one of my students as a research assistant for £2185 per month, plus University overhead fees.

Opportunities

The initial prototype, especially following option (a), is unlikely to lead to patentable technologies, as the basic principles have applied long ago for noise suppression in digital TV receivers. Likewise, the reconstruction of video sync pulses was mentioned at least as early as 1985 (van Eck) and is implemented in commercial products. The proposed initial development is based on well-known principles, but for a parameter range that is not supported in existing products. There might potentially be some patentable opportunities for the full-featured design, especially should we be able to come up with a practically working automatic frequency tracking mechanism.

The main intellectual property produced in the project would be the circuit programmed into the FPGA. Should this project make it to a very advanced stage, with most of the features described above working reliably, then the project could be wrapped up in a low-volume specialist market product that might be of interest to some customers in the Tempest and signals intelligence business.

References

[1] Markus G. Kuhn: *Compromising emanations: eavesdropping risks of computer displays*. Technical Report UCAM-CL-TR-577, University of Cambridge, Computer Laboratory, December 2003.