





**Figure 2: Experiment in Lab #1 with transmitter and receiver separated by 55ft. By applying a filter with passband of <67.28kHz, 67.34kHz>, the amplitude of filtered voltage signals acquired by the receiver matches the transmitter’s current amplitude.**

normal programs, but no special privileges are assumed by NoDE. The receiver, on the other hand, can be any innocuous-looking device that is plugged into an outlet in the same building’s power network as the transmitter. The receiver needs an ADC (analog to digital converter) for digitizing its received voltage and can be easily hidden inside a laptop/cellphone charger. Moreover, the receiver can be located in a distant room different than the transmitter.

### 3 NODE: EXTRACTING TRANSMITTER’S CURRENT FROM RECEIVER VOLTAGE

All desktop computers today are mandated to have built-in power PFC circuits in their power supply units to reduce harmonics. Importantly, these PFC circuits result in prominent high-frequency current ripples between 40kHz and 150kHz, whose amplitude changes with the computer’s power consumption — the higher power consumption, the taller ripples, and vice-versa. These high-frequency current ripples also produce high-frequency voltage ripples at other power outlets, which are referred to as switching noises. Thus, by properly filtering the received voltage signals at a power outlet, switching noises can be retained and the receiver is able to successfully extract information about the transmitter’s modulated current amplitude. Further, for different computers, the switching noise frequencies are typically different, which allows simultaneous data exfiltration attacks from multiple computers by a single receiver without strong interference.

We empirically validate the feasibility of extracting the transmitter’s current amplitude information. We first show in Fig. 2(a) the transmitter’s CPU load and current amplitude. Next, Fig. 2(b) shows the frequency components of the voltage signal at the receiver. We see large frequency components between 40kHz and 80kHz due to different computers’ PFC switching operations, and the components around 67.3kHz are caused by our transmitter. The temporal variation of the power spectral density spikes created by the transmitter is shown in the frequency spectrum in Fig. 2(c), where we can easily identify the transmitter’s high current periods. Next, we filter the collected voltage signal with a passband of <67.28kHz, 67.34kHz> and show the filtered voltage signal in Fig. 2(d) where the filtered voltage signal closely resembles the current ripples.

### 4 EVALUATION

We evaluate NoDE using seven computers with different hardware and software configurations at five different locations with various numbers of computers (e.g., Lab#1 has 30+ computers) in two different buildings. Table 1 lists the transmitter computers and their

**Table 1: Summary of Experiments on Different Computers.**

Transmitting Computer	Location	TX-RX Distance	Bit Error Rate	Bits Per Second
Dell Optiplex 9020, Windows 10	Lab #1 (Building A)	~55 feet	0.0%	28.48
Dell PowerEdge R630, Ubuntu 14.04	Office (Building B)	~90 feet	0.0%	28.48
Dell XPS 8920, Windows 10	Lab #1 (Building A)	~55 feet	0.0%	28.48
Acer G3-710, Ubuntu 16.04	Lab #2 (Building A)	~20 feet	10.1%	25.60
Custom Built with Corsair, Windows 10	Lab #1 (Building A)	~55 feet	8.1%	26.17
Custom-Built with EVGA, Windows	Lab #3 (Building A)	~15 feet	9.2%	25.85
Apple iMac Model A1419 (27-inch)	Lab #1 (Building A)	~55 Feet	16% (50ms/sym)	15.79
			2% (100ms/sym)	9.21

distances from the receiver at different locations along with the resulting bit error and bit transfer rates. We demonstrate NoDE can simultaneously exfiltrate data from four transmitters using a single receiver. We also test NoDE’s transmission accuracy under different background applications and tasks (e.g., MS Word), different number of CPU cores used by NoDE, and different data frame settings. We generally observe very low error and high bit transmission rates across different settings.

### 5 DEFENSE

To identify possible safeguards against NoDE’s data exfiltration, we evaluate different hardware and software based defense approaches such as using a UPS to mask transmitter’s power variation, adding power line noise filter to restrict PFC switching noise from entering the power network, and suppressing malware activity by randomizing computer’s power consumption. Based on our study, we recommend the installation of power noise filters as a hardware-based defense and power randomization as a software-based technique.

### ACKNOWLEDGMENTS

This work was supported in part by the U.S. NSF under grants CNS-1551661, ECCS-1610471, and CNS-1910208.

### REFERENCES

[1] Zhihui Shao, Mohammad A. Islam, and Shaolei Ren. 2020. Your Noise, My Signal: Exploiting Switching Noise for Stealthy Data Exfiltration from Desktop Computers. *Proc. ACM Meas. Anal. Comput. Syst.* 4, 1, Article Article 7 (March 2020), 39 pages. <https://doi.org/10.1145/3379473>